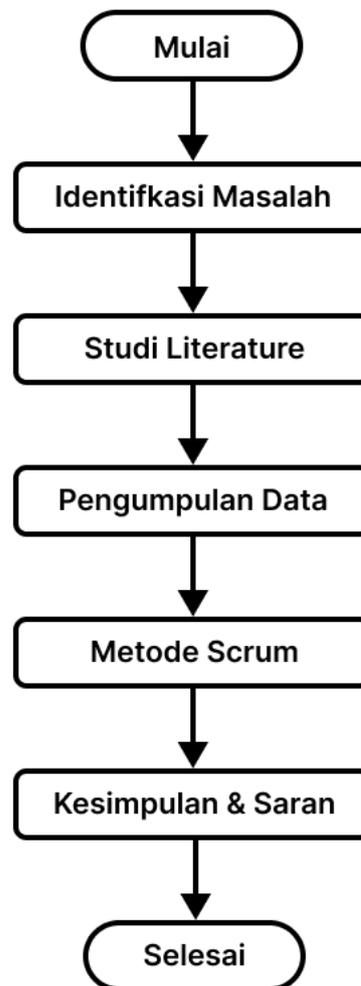


## BAB III

### METODOLOGI PENELITIAN

#### 1.1 Alur Penelitian

Berikut alur penelitian atau konsep untuk gambaran dari penelitian yang akan dilakukan. Pada perancangan sistem menggunakan *SCRUM* diperlukan panduan dalam proses pengerjaan penelitian skripsi ini. Penjabaran alur penelitian tersebut dapat dilihat dapat dilihat pada gambar dibawah ini:



**Gambar 3.1** Alur Penelitian

## 1.2 Pengumpulan Data

Pada tahap ini, dilakukan pengumpulan data yang merupakan tahap untuk memperoleh informasi dan data yang berhubungan dengan penelitian. Pengumpulan data dilakukan dengan mencari data melalui *online repository* yang telah disediakan oleh pemerintah Kota Bandar Lampung (<https://bandarlampungkota.go.id>) dan web Google Maps (<https://maps.google.com>). Hal ini dilakukan untuk mempermudah dan mempersingkat waktu dalam pencarian data pendukung. Data-data tersebut berupa data pos pemadam kebakaran, data kejadian kebakaran, serta data jaringan jalan wilayah Kota Bandar Lampung. Berikut data lokasi pos pemadam kebakaran yang diperoleh:

**Tabel 3.1** Data Lokasi Pos Pemadam Kebakaran

| No | Nama Pos Pemadam        | Alamat  | Latitude   | Longitude   |
|----|-------------------------|---|------------|-------------|
| 1  | Dinas Pemadam Kebakaran | Jl. Kapten Tendean No.2, Palapa, Kec. Tj. Karang Pusat, Kota Bandar Lampung, Lampung 35116      | -5.421126  | 105.2537369 |
| 2  | Teluk Betung Timur      | G6JW+66R, Sukamaju, Kec. Telukbetung Timur, Kota Bandar Lampung, Lampung 35231                  | -5.469388  | 105.2449756 |
| 3  | Labuhan Ratu            | J7R3+4J5, Kp. Baru, Kec. Kedaton, Kota Bandar Lampung, Lampung 35141                            | -5.3597498 | 105.2535567 |
| 4  | Way Halim               | J78G+GJF, Jl. Griya Utama, Way Halim Permai, Kec. Way Halim, Kota Bandar Lampung, Lampung 35135 | -5.383696  | 105.2759887 |

### 1.3 User Story

Pada tahapan ini menjelaskan siapa saja yang akan menjadi pengguna sistem beserta tugas dan tujuannya. Berikut adalah daftar *user story* yang berhasil dikumpulkan.

**Tabel 3.2** User Story

| No | Pengguna                      | User Story   |
|----|-------------------------------|--|
| 1  | Admin                         | Sebagai admin, user dapat mengelola data seperti data petugas pemadam kebakaran, data lokasi pemadam kebakaran, dan laporan kebakaran. |
| 2  | Masyarakat                    | Sebagai masyarakat, user dapat melakukan registrasi kemudian melaporkan kejadian kebakaran dan melihat riwayat laporan.                |
| 3  | Operator Pemadam<br>Kebakaran | Sebagai operator pemadam kebakaran, user dapat menerima laporan dan melihat riwayat laporan.   |

### 1.4 Product Backlog

Pada proses tahapan ini peneliti melakukan pengumpulan kebutuhan yang akan dilakukan melalui daftar prioritas kebutuhan sistem. Berikut ini merupakan *product backlog* yang diperoleh dari hasil *user story* yang sudah diidentifikasi sebelumnya.

**Tabel 3.3** Product Backlog

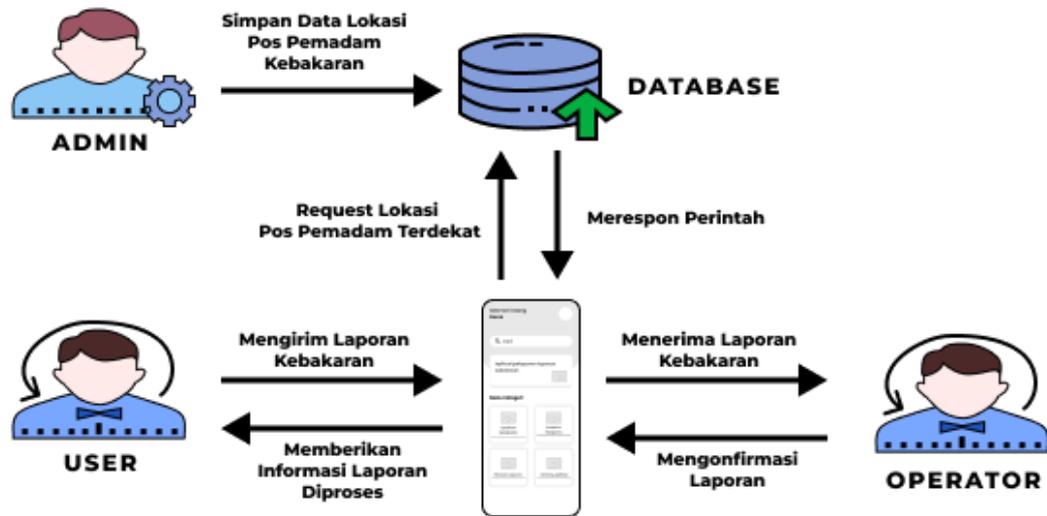
| No | Nama Backlog                      | Prioritas |
|----|-----------------------------------|-----------|
| 1  | Kebutuhan Dokumen dan Data        | Sedang    |
| 2  | Perancangan Sistem dan Basis Data | Tinggi    |
| 3  | Halaman Login Admin               | Tinggi    |
| 4  | Dashboard Admin                   | Sedang    |
| 5  | Halaman Profil Admin              | Rendah    |

|    |   |        |
|----|---|--------|
| 6  | Halaman Kelola Petugas Operator               | Tinggi |
| 7  | Halaman Lokasi                                | Tinggi |
| 8  | Halaman Data User                             | Rendah |
| 9  | Halaman Data Laporan Kebakaran                | Rendah |
| 10 | Menu Pengelolaan Data-Data Pada Website Admin | Sedang |
| 11 | Struktur Menu Pada Halaman Utama              | Sedang |
| 12 | Halaman Daftar User (Masyarakat)              | Tinggi |
| 13 | Halaman Login User (Masyarakat)               | Tinggi |
| 14 | Dashboard Dan Profile User (Masyarakat)       | Sedang |
| 15 | Halaman Utama (Menu)                          | Tinggi |
| 16 | Halaman Laporkan Kebakaran                    | Tinggi |
| 17 | Halaman Riwayat Laporan (Masyarakat)          | Sedang |
| 18 | Halaman Tentang Aplikasi                      | Rendah |
| 19 | Halaman Login Petugas Operator                | Tinggi |
| 20 | Halaman Laporan Kebakaran                     | Sedang |
| 21 | Halaman Riwayat Laporan (Petugas Operator)    | Rendah |
| 22 | Halaman Profil Petugas Operator               | Sedang |

Pada *product backlog* yang terdapat pada tabel diatas, aktivitas nomor 1 dan 2 ditunjukkan untuk mengidentifikasi dan merancang sistem informasi berdasarkan dokumen dan data yang telah dikumpulkan. Selanjutnya aktivitas nomor 3 dan seterusnya berfokus pada pengembangan sistem berbasis mobile android dan website untuk admin yang dalam implementasinya menggunakan XML dan bahasa Java untuk android serta bahasa PHP untuk website admin.

Berikut perancangan sistem yang meliputi tahap desain sistem. Adapun sistem yang akan dibangun antara lain *flowchart*, rancangan algoritma, *Use Case Diagram*, *Activity Diagram*, *Class Diagram* dan terdapat gambaran antarmuka (*interface design*) sesuai dengan kebutuhan yakni sebagai berikut:

### 1.4.1 Diagram Blok Sistem

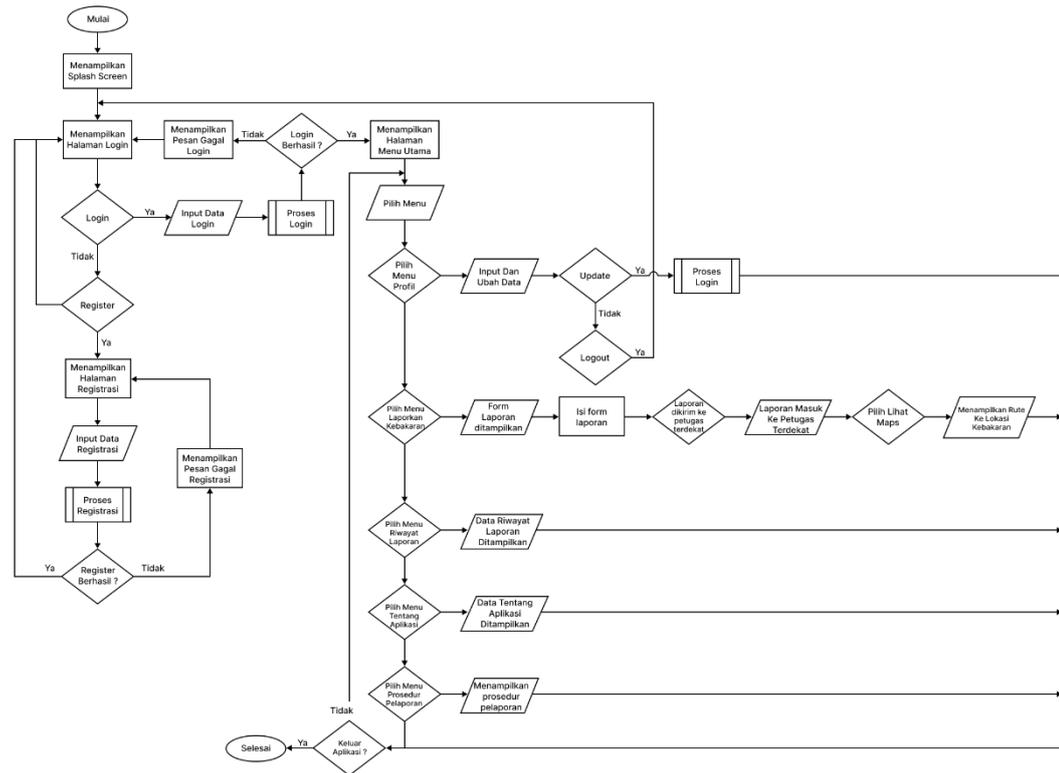


Gambar 3.2 Diagram Blok Sistem

Alur kerja dari diagram blok sistem adalah sebagai berikut. User mengirim laporan melalui aplikasi, kemudian aplikasi merequest data lokasi pemadam kebakaran terdekat ke database. Lalu database merespon perintah dengan meneruskan laporan ke operator pos pemadam kebakaran terdekat. Kemudian operator menerima laporan kebakaran melalui aplikasi dan mengonfirmasi laporan tersebut. Setelah laporan dikonfirmasi oleh operator, maka aplikasi akan memberi notifikasi kepada user bahwa laporan yang dikirim telah diproses.

### 1.4.2 Flowchart Sistem Aplikasi

Berikut merupakan *flowchart* sistem aplikasi yang akan digunakan pada aplikasi *smart service* layanan kebakaran:



**Gambar 3.3** Flowchart Sistem Aplikasi

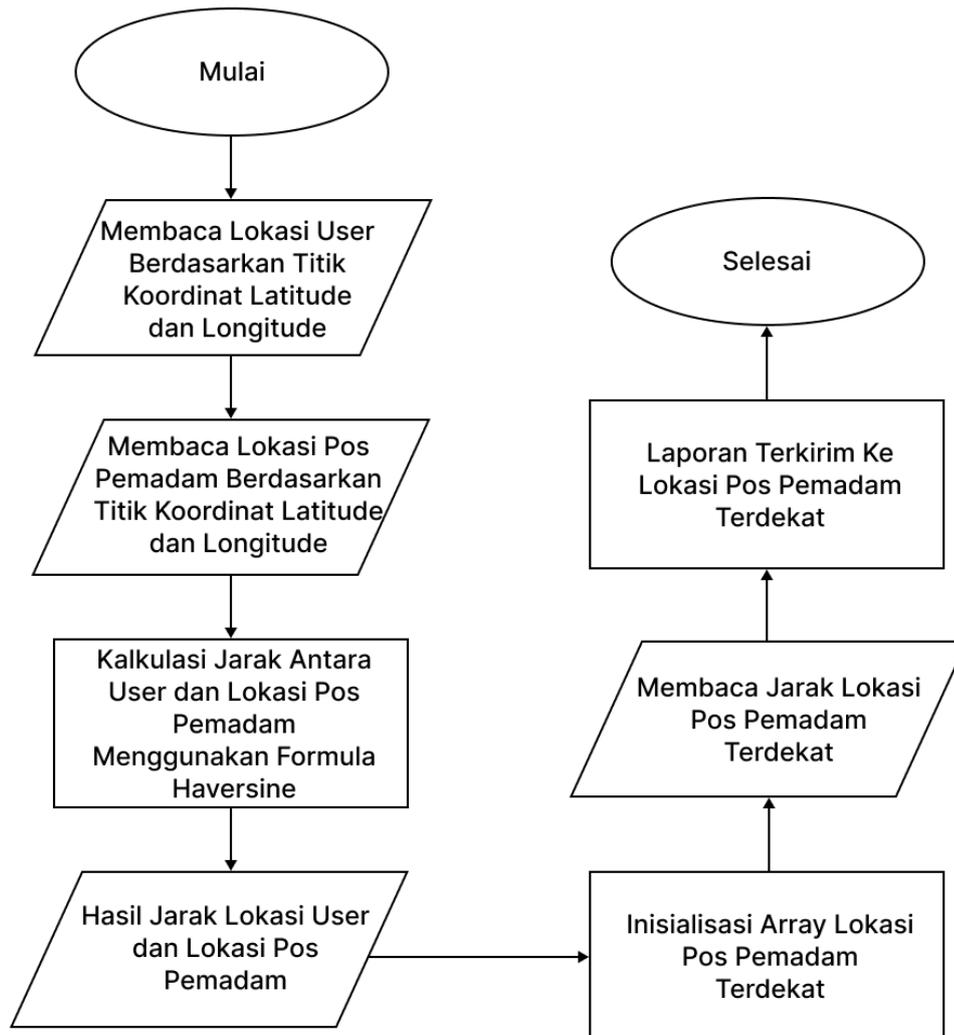
Pada gambar flowchart diatas dapat dilihat alur aplikasi secara garis besar. Untuk mempermudah pemahaman alur dari aplikasi, dibuatlah flowchart secara keseluruhan dan flowchart pendukung proses dari aplikasi ini. Alur dari aplikasi ini sama seperti aplikasi sejenis pada umumnya, yaitu dimulai dengan halaman login dan tampilan menu dimulai dari menu utama kemudian menuju menu yang lebih khusus atau spesifik.

Dari gambar 3.3 dapat dilihat bahwa pertama kali yang ditampilkan adalah splash screen logo dari aplikasi, selanjutnya form login akan ditampilkan dengan inputan berupa username atau email dan password. Jika belum memiliki akun maka akan diarahkan ke halaman register yang berisi form pendaftaran seperti nama, no hp, email, alamat dan password.

Setelah proses login berhasil maka akan ditampilkan menu utama dari aplikasi yaitu profil, laporan kebakaran, riwayat laporan, tentang aplikasi dan prosedur pelaporan. Pada menu laporan kebakaran akan menampilkan form yang wajib diisi oleh user. Setelah mengisi form tersebut user menekan tombol kirim

kemudian laporan akan masuk ke pos pemadam kebakaran terdekat. Setelah itu operator pemadam akan menerima laporan dan membuka detail laporan. Kemudian operator menekan lokasi yang dikirim oleh pengirim, maka akan diarahkan ke aplikasi Google Maps untuk mendapatkan rute ke lokasi kebakaran.

### 1.4.3 Rancangan Algoritma Haversine Formula



**Gambar 3.4** *Flowchart Haversine*

Studi Kasus:

Pada studi kasus ini akan menghitung jarak terdekat dari lokasi kebakaran menuju pos pemadam terdekat. Sebagai contoh, pada studi kasus ini menggunakan lokasi Institut Informatika dan Bisnis Darmajaya sebagai lokasi kebakaran dengan titik koordinat -5.3774078,105.2496419.

## a. Studi Kasus 1

Pada studi kasus pertama akan menghitung jarak dari Institut Informatika dan Bisnis Darmajaya (koordinat -5.3774078,105.2496419) ke Dinas Pemadam Kebakaran (koordinat -5.421126, 105.2537369).

Penyelesaian

Rumus Haversine:

$$d = 2r \cdot \arcsin \left( \sqrt{\sin^2 \left( \frac{Lat2 - Lat1}{2} \right) + \cos(Lat2) \cdot \cos(Lat1) \cdot \sin^2 \left( \frac{Lon2 - Lon1}{2} \right)} \right)$$

$$d = 2 \cdot 6371 \cdot \arcsin(\sqrt{1,45552E-07 + 0,991145877 \cdot 1,27703E-09})$$

$$d = 12742 \cdot \arcsin(\sqrt{1,46818E-07})$$

$$d = 12742 \cdot \arcsin(0,000383168)$$

$$d = 12742 \cdot 0,000383168$$

$$d = 4,88 \text{ km}$$

## b. Studi Kasus 2

Pada studi kasus kedua akan menghitung jarak dari Institut Informatika dan Bisnis Darmajaya (koordinat -5.3774078,105.2496419) ke Pos Pemadam di Teluk Betung Timur (koordinat -5.469388, 105.2449756).

Penyelesaian

Rumus Haversine:

$$d = 2r \cdot \arcsin \left( \sqrt{\sin^2 \left( \frac{Lat2 - Lat1}{2} \right) + \cos(Lat2) \cdot \cos(Lat1) \cdot \sin^2 \left( \frac{Lon2 - Lon1}{2} \right)} \right)$$

$$d = 2 \cdot 6371 \cdot \arcsin(\sqrt{6,44293E-07 + 0,991066296 \cdot 1,65821E-09})$$

$$d = 12742 \cdot \arcsin(\sqrt{6,45936E-07})$$

$$d = 12742 \cdot \arcsin(0,000803702)$$

$$d = 12742 \cdot 0,000803702$$

$$d = 10,24 \text{ km}$$

## c. Studi Kasus 3

Pada studi kasus ketiga akan menghitung jarak dari Institut Informatika dan Bisnis Darmajaya (koordinat -5.3774078,105.2496419) ke Pos Pemadam di Labuhan Ratu (koordinat -5.3597498, 105.2449756).

Penyelesaian

Rumus Haversine:

$$d = 2r \cdot \arcsin \left( \sqrt{\sin^2 \left( \frac{Lat2 - Lat1}{2} \right) + \cos(Lat2) \cdot \cos(Lat1) \cdot \sin^2 \left( \frac{Lon2 - Lon1}{2} \right)} \right)$$

$$d = 2 \cdot 6371 \cdot \arcsin(\sqrt{2,37453E-08 + 0,991246067 \cdot 1,16712-09})$$

$$d = 12742 \cdot \arcsin(\sqrt{2,49022E-08})$$

$$d = 12742 \cdot \arcsin(0,000157804)$$

$$d = 12742 \cdot 0,000157804$$

$$d = 2,01 \text{ km}$$

d. Studi Kasus 4

Pada studi kasus keempat akan menghitung jarak dari Institut Informatika dan Bisnis Darmajaya (koordinat -5.3774078,105.2496419) ke Pos Pemadam di Way Halim (koordinat -5.383696, 105.2759887).

Penyelesaian

Rumus Haversine:

$$d = 2r \cdot \arcsin \left( \sqrt{\sin^2 \left( \frac{Lat2 - Lat1}{2} \right) + \cos(Lat2) \cdot \cos(Lat1) \cdot \sin^2 \left( \frac{Lon2 - Lon1}{2} \right)} \right)$$

$$d = 2 \cdot 6371 \cdot \arcsin(\sqrt{3,01125E-09 + 0,991207113 \cdot 5,28628-08})$$

$$d = 12742 \cdot \arcsin(\sqrt{5,54093E-08})$$

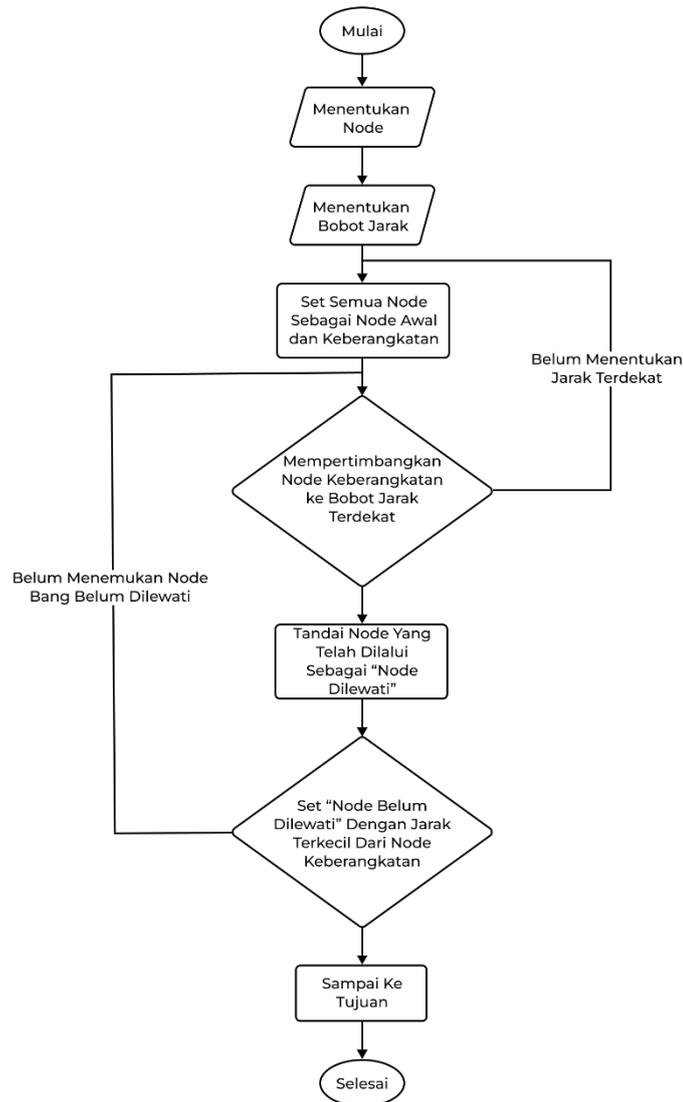
$$d = 12742 \cdot \arcsin(0,000235392)$$

$$d = 12742 \cdot 0,000235392$$

$$d = 2,99 \text{ km}$$

Setelah dilakukan perhitungan, maka di dapatkan hasil pos pemadam kebakaran terdekat dari Institut Informatika dan Bisnis Darmajaya yaitu pos pemadam kebakaran di Labuhan Ratu dengan jarak 2,01 km. Dengan ini maka laporan kebakaran akan terkirim ke pos pemadam kebakaran di Labuhan Ratu.

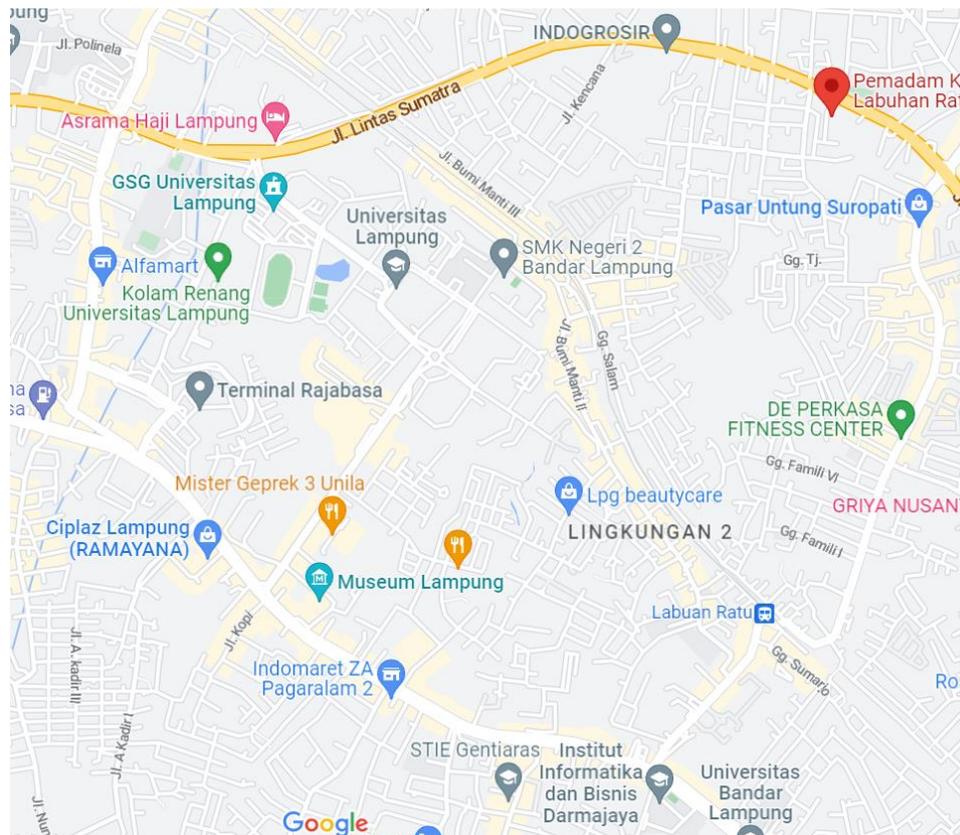
### 1.4.4 Rancangan Algoritma Dijkstra



**Gambar 3.5** Flowchart Dijkstra

#### a. Merumuskan Titik Awal

Setelah menemukan lokasi pos pemadam terdekat menggunakan formula *haversine*, selanjutnya yaitu merumuskan titik awal. Peneliti mengambil sampel dari *Google Maps* untuk mengetahui peta yang diinginkan. Setelah mengetahui petanya, langkah selanjutnya adalah menentukan titik dimana lokasi kebakaran berada di peta.

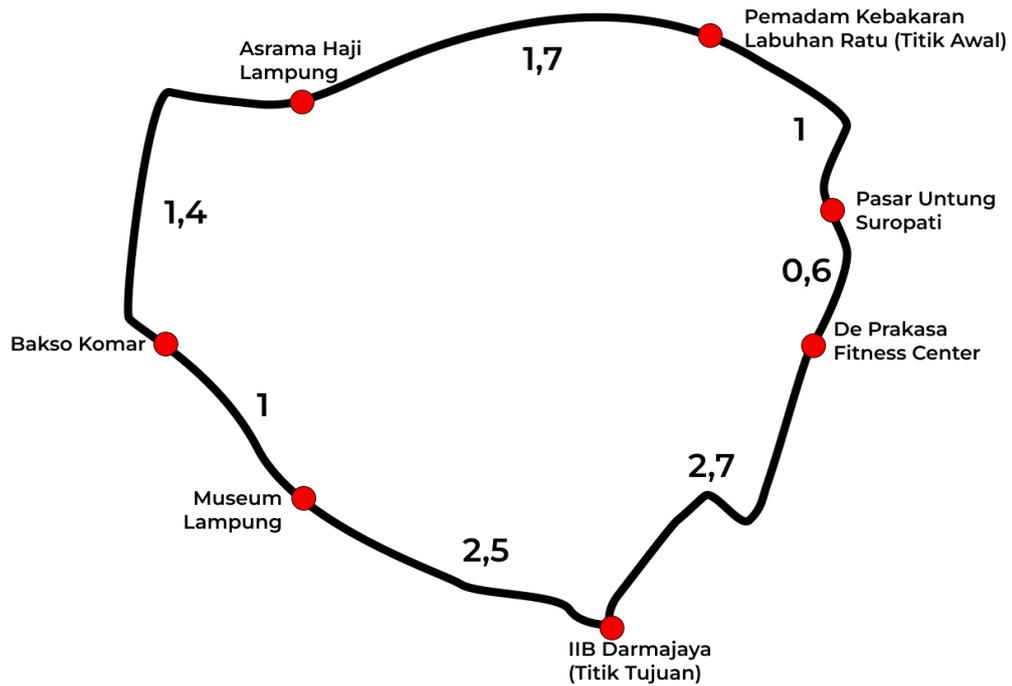


**Gambar 3.6** Peta Lokasi Pemadam Kebakaran Labuhan Ratu

Setelah mengetahui titik lokasi, peneliti dapat menggambarannya dalam bentuk graf agar memudahkan dalam melakukan perhitungan pencarian lintasan atau rute terpendek. Tujuan dari peta lokasi ini adalah untuk mengetahui manakah jalan antar titik yang lebih efektif dan lebih cepat. Selain ini, pemilihan rute juga didasarkan pada kondisi jalan yang dilalui layak atau tidak dan mungkin sering terjadi kemacetan. Dengan pemilihan ini maka peneliti bisa mengetahui jalan yang paling efektif dan paling dekat antar titik.

#### **b. Menentukan Bobot Dari Graf**

Langkah selanjutnya yaitu menarik garis sebagai lintasan agar dapat dibentuk dalam graf dan kemudian mengukur panjang lintasan yang ada.



**Gambar 3.7** Graf Berbobot

Langkah selanjutnya adalah membuat tabel dari graf untuk mengetahui data yang didapatkan dan kemudian dihitung menggunakan algoritma *Dijkstra*. Dengan tabel ini, langkah untuk menghitung lintasan terpendek akan menjadi lebih mudah.

**Tabel 3.4** Ringkasan Antar Titik dan Jaraknya

| Titik 1                        | Titik 2                   | Jarak Antar Titik (Km) |
|--------------------------------|---------------------------|------------------------|
| Pemadam Kebakaran Labuhan Ratu | Pasar Untung Suropati     | 1                      |
| Pemadam Kebakaran Labuhan Ratu | Asrama Haji Lampung       | 1,7                    |
| Pasar Untung Suropati          | De Prakasa Fitness Center | 0,6                    |
| Asrama Haji Lampung            | Bakso Komar               | 1,4                    |
| De Prakasa Fitness Center      | IIB Darmajaya             | 2,7                    |
| Bakso Komar                    | Museum Lampung            | 1                      |
| Museum Lampung                 | IIB Darmajaya             | 2,5                    |

### c. Mencari Lintasan Terpendek

Selanjutnya yaitu mencari lintasan terpendek menuju IIB Darmajaya sebagai contoh lokasi kebakaran. Sebelum mencari lintasan terpendek, terlebih dahulu menentukan titik awal sebagai titik permulaan perhitungan. Setelah menentukan titik awal, selanjutnya menentukan titik akhir sebagai tujuan selesainya perhitungan. Dalam kasus penelitian ini, titik awalnya adalah Pemadam Kebakaran Labuhan Ratu. Peneliti mencari rute terpendek dalam menuju node-node pada gambar dengan titik awal Pemadam Kebakaran Labuhan Ratu dan titik tujuannya adalah IIB Darmajaya. Sebelum melakukan perhitungan, terlebih dahulu membuat matriks keterkaitan antar titik.

**Tabel 3.5** Matriks Berhubungan

|                                | Pemadam Kebakaran Labuhan Ratu | Pasar Untung Suropati | Asrama Haji Lampung | De Prakasa Fitness Center | Bakso Komar | Museum Lampung | IIB Darmajaya |
|--------------------------------|--------------------------------|-----------------------|---------------------|---------------------------|-------------|----------------|---------------|
| Pemadam Kebakaran Labuhan Ratu | 0                              | 1                     | 1                   | 0                         | 0           | 0              | 0             |
| Pasar Untung Suropati          | 1                              | 0                     | 0                   | 1                         | 0           | 0              | 0             |
| Asrama Haji Lampung            | 1                              | 0                     | 0                   | 0                         | 1           | 0              | 0             |
| De Prakasa Fitness Center      | 0                              | 1                     | 0                   | 0                         | 0           | 0              | 1             |
| Bakso Komar                    | 0                              | 0                     | 1                   | 0                         | 0           | 1              | 0             |
| Museum Lampung                 | 0                              | 0                     | 0                   | 0                         | 1           | 0              | 1             |
| IIB Darmajaya                  | 0                              | 0                     | 0                   | 1                         | 0           | 1              | 0             |

Matriks keterhubungan ini bertujuan untuk mempermudah mengetahui titik mana saja yang saling terhubung. Nilai 1 pada matriks

keterhubungan menandakan bahwa antar titik satu dengan titik lainnya saling terhubung. Sedangkan nilai 0 menandakan titik satu dengan yang lain tidak saling terhubung.

Setelah mengetahui mana saja titik yang saling terhubung, langkah selanjutnya yaitu menghitung secara manual. Pilih titik awal Pemadam Kebakaran Labuhan Ratu dan cari titik terdekat dengannya, yaitu Pasar Untung Suropati dan Asrama Haji Lampung dengan  $\text{Bobot}_{(\text{Pemadam Kebakaran Labuhan Ratu})} = 0$ ,  $\text{Bobot}_{(\text{Pasar Untung Suropati})} = \infty$  dan  $\text{Bobot}_{(\text{Asrama Haji Lampung})} = \infty$ , maka langkah perhitungannya adalah sebagai berikut:

$$\text{Bobot}_{(\text{Pasar Untung Suropati})} = (\text{Bobot}_{(\text{Pasar Untung Suropati})}, \text{Bobot}_{(\text{Pemadam Kebakaran Labuhan Ratu})} + \text{Jarak}_{(\text{Pemadam Kebakaran Labuhan Ratu ke Pasar Untung Suropati})})$$

$$\text{Bobot}_{(\text{Pasar Untung Suropati})} = (\infty, 0 + 1)$$

$$\text{Bobot}_{(\text{Pasar Untung Suropati})} = (\infty, 1)$$

$$\text{Bobot}_{(\text{Pasar Untung Suropati})} = 1$$

$$\text{Bobot}_{(\text{Asrama Haji Lampung})} = (\text{Bobot}_{(\text{Asrama Haji Lampung})}, \text{Bobot}_{(\text{Pemadam Kebakaran Labuhan Ratu})} + \text{Jarak}_{(\text{Pemadam Kebakaran Labuhan Ratu ke Asrama Haji Lampung})})$$

$$\text{Bobot}_{(\text{Asrama Haji Lampung})} = (\infty, 0 + 1,7)$$

$$\text{Bobot}_{(\text{Asrama Haji Lampung})} = (\infty, 1,7)$$

$$\text{Bobot}_{(\text{Asrama Haji Lampung})} = 1,7$$

Nilai  $\text{Bobot}_{(\text{Asrama Haji Lampung})} > \text{Bobot}_{(\text{Pasar Untung Suropati})}$  sehingga Pasar Untung Suropati terpilih sebagai titik selanjutnya. Titik terdekat dari Pasar Untung Suropati adalah De Prakasa Fitness Center dengan  $\text{Bobot}_{(\text{De Prakasa Fitness Center})} = \infty$ .

$$\text{Bobot}_{(\text{De Prakasa Fitness Center})} = (\text{Bobot}_{(\text{De Prakasa Fitness Center})}, \text{Bobot}_{(\text{Pasar Untung Suropati})} + \text{Jarak}_{(\text{De Prakasa Fitness Center ke Pasar Untung Suropati})})$$

$$\text{Bobot}_{(\text{De Prakasa Fitness Center})} = (\infty, 1 + 0,6)$$

$$\text{Bobot}_{(\text{De Prakasa Fitness Center})} = (\infty, 1,6)$$

$$\text{Bobot}_{(\text{De Prakasa Fitness Center})} = 1,6$$

Karena hanya ada satu titik, maka De Prakasa Fitness Center terpilih sebagai titik selanjutnya. Titik terdekat dari De Prakasa Fitness Center adalah IIB Darmajaya yang merupakan titik tujuan akhir.  $\text{Bobot}_{(\text{IIB Darmajaya})} = \infty$ .

$$\text{Bobot}_{(\text{IIB Darmajaya})} = (\text{Bobot}_{(\text{IIB Darmajaya})}, \text{Bobot}_{(\text{De Prakasa Fitness Center})} + \text{Jarak}_{(\text{De Prakasa Fitness Center ke IIB Darmajaya})})$$

$$\text{Bobot}_{(\text{IIB Darmajaya})} = (\infty, 1,6 + 2,7)$$

$$\text{Bobot}_{(\text{IIB Darmajaya})} = (\infty, 4,3)$$

$$\text{Bobot}_{(\text{IIB Darmajaya})} = 4,3$$

IIB Darmajaya terpilih sebagai titik selanjutnya.

Perhitungan diatas memperoleh titik dimana saja yang dikunjungi dalam mencari lintasan terpendek. Setelah mengetahui titik mana saja yang dikunjungi, maka terbentuklah lintasan terpendek dari titik awal Pemadam Kebakaran Labuhan Ratu menuju titik akhir IIB Darmajaya. Lintasan ini yang kemudian dijadikan sebagai hasil pencarian lintasan terpendek dan ditetapkan sebagai lintasan terpendek perhitungan sementara. Hasil lintasan terpendek yang didapatkan sebagai titik akhir yaitu : Pemadam Kebakaran Labuhan Ratu  $\rightarrow$  Pasar Untung Suropati  $\rightarrow$  De Prakasa Fitness Center  $\rightarrow$  IIB Darmajaya yang memperoleh 4 titik yang dikunjungi dengan jarak tempuh yaitu 4,3 km.

Sistem perhitungan rute masih ada titik yang belum terhitung jaraknya, jadi harus mencari rute terpendek untuk perhitungan kedua. Untuk perhitungan kedua, mencari rute terpendek dimulai dari titik Pemadam Kebakaran Labuhan Ratu menuju titik IIB Darmajaya. Titik yang sudah dilewati di perhitungan pertama diabaikan karena sudah dilewati. Titik terdekat dari Pemadam Kebakaran Labuhan Ratu adalah Pasar Untun Suropati dan Asrama Haji Lampung. Karena Pasar Untung Suropati sudah dilewati pada perhitungan pertama, maka titik yang digunakan adalah Asrama Haji Lampung dengan  $\text{Bobot}_{(\text{Pemadam Kebakaran Labuhan Ratu})} = 0$ ,  $\text{Bobot}_{(\text{Asrama Haji Lampung})} = \infty$  maka langkah perhitungannya sebagai berikut:

$$\text{Bobot}(\text{Asrama Haji Lampung}) = (\text{Bobot}(\text{Asrama Haji Lampung}), \text{Bobot}(\text{Pemadam Kebakaran Labuhan Ratu}) + \text{Jarak}(\text{Pemadam Kebakaran Labuhan Ratu ke Asrama Haji Lampung}))$$

$$\text{Bobot}(\text{Asrama Haji Lampung}) = (\infty, 0 + 1,7)$$

$$\text{Bobot}(\text{Asrama Haji Lampung}) = (\infty, 1,7)$$

$$\text{Bobot}(\text{Asrama Haji Lampung}) = 1,7$$

Karena hanya ada satu titik, maka Asrama Haji Lampung terpilih sebagai titik selanjutnya. Titik terdekat dari Asrama Haji Lampung adalah Bakso Komar dengan  $\text{Bobot}(\text{Bakso Komar}) = \infty$ .

$$\text{Bobot}(\text{Bakso Komar}) = (\text{Bobot}(\text{Bakso Komar}), \text{Bobot}(\text{Asrama Haji Lampung}) + \text{Jarak}(\text{Asrama Haji Lampung ke Bakso Komar}))$$

$$\text{Bobot}(\text{Bakso Komar}) = (\infty, 1,7 + 1,4)$$

$$\text{Bobot}(\text{Bakso Komar}) = (\infty, 3,1)$$

$$\text{Bobot}(\text{Bakso Komar}) = 3,1$$

Karena hanya ada satu titik, maka Bakso Komar terpilih sebagai titik selanjutnya. Titik terdekat dari Bakso Komar adalah Museum Lampung dengan  $\text{Bobot}(\text{Museum Lampung}) = \infty$ .

$$\text{Bobot}(\text{Museum Lampung}) = (\text{Bobot}(\text{Museum Lampung}), \text{Bobot}(\text{Bakso Komar}) + \text{Jarak}(\text{Bakso Komar ke Museum Lampung}))$$

$$\text{Bobot}(\text{Museum Lampung}) = (\infty, 3,1 + 1)$$

$$\text{Bobot}(\text{Museum Lampung}) = (\infty, 4,1)$$

$$\text{Bobot}(\text{Museum Lampung}) = 4,1$$

Karena hanya ada satu titik, maka Museum Lampung terpilih sebagai titik selanjutnya. Titik terdekat dari Museum Lampung adalah IIB Darmajaya yang merupakan titik tujuan akhir dengan  $\text{Bobot}(\text{IIB Darmajaya}) = \infty$ .

$$\text{Bobot}(\text{IIB Darmajaya}) = (\text{Bobot}(\text{IIB Darmajaya}), \text{Bobot}(\text{Museum Lampung}) + \text{Jarak}(\text{Museum Lampung ke IIB Darmajaya}))$$

$$\text{Bobot}(\text{IIB Darmajaya}) = (\infty, 4,1 + 2,5)$$

$$\text{Bobot}(\text{IIB Darmajaya}) = (\infty, 6,6)$$

$$\text{Bobot}_{(\text{IIB Darmajaya})} = 6,6$$

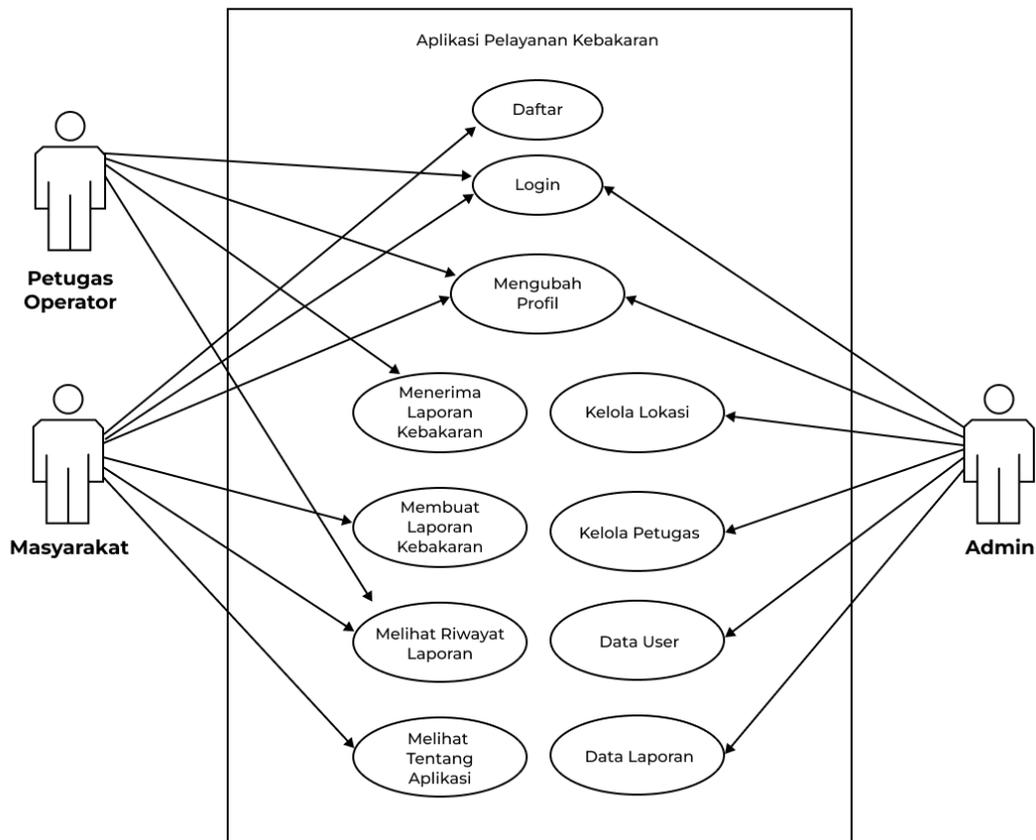
IIB Darmajaya terpilih sebagai titik selanjutnya dan menjadi tempat tujuan terakhir.

Diperoleh titik mana saja yang dikunjungi dalam pencarian rute terpendek pada perhitungan diatas. Perhitungan diatas memperoleh hasil rute terpendek dari titik awal Pemadam Kebakaran Labuhan Ratu menuju titik akhir IIB Darmajaya. Hasil perhitungan kedua yang dilalui yaitu : Pemadam Kebakaran Labuhan Ratu → Asrama Haji Lampung → Bakso Komar → Museum Lampung → IIB Darmajaya yang mendapatkan 5 titik dengan jarak 6,6 km.

Hasil yang didapatkan setelah melakukan perhitungan pencarian rute terpendek menggunakan algoritma *Dijkstra* telah selesai karena semua titik telah dikunjungi dan menghasilkan rute terpendek dari awal menuju akhir yaitu dimulai dari Pemadam Kebakaran Labuhan Ratu → Pasar Untung Suropati → De Prakasa Fitness Center → IIB Darmajaya dengan jarak tempuh 4,3 km.

#### **1.4.5 Use Case Diagram**

*Use Case Diagram* dibuat untuk mengidentifikasi aktor-aktor yang berperan dalam aplikasi pelayanan kebakaran berbasis android dan menggambarkan terkait apa saja yang dapat dilakukan oleh masing-masing aktor dalam sistem.



**Gambar 3.8** Use Case Diagram

Pada gambar diatas dapat dilihat bahwa pada sistem yang akan dikembangkan terdapat tiga aktor yaitu admin, petugas operator dan user masyarakat dengan aktivitas yang dilakukan oleh masing-masing aktor. Berikut adalah penjelasan aktivitas yang dilakukan oleh masing-masing aktor:

1. Admin bertanggung jawab dalam mengelola data. Pada aktor admin terdapat 6 fungsionalitas yaitu login sebagai admin, mengubah profil admin, kelola lokasi pemadam yang berfungsi untuk menambahkan lokasi pemadam jika ada lokasi baru, kelola data petugas operator pemadam yang berfungsi untuk menambah data petugas pemadam jika ada petugas operator baru, kelola data user dan mengelola laporan kebakaran yang dikirim oleh masyarakat. Admin mengelola data hanya melalui website yang terhubung dengan database dan juga aplikasi dari petugas operator dan user.
2. Pada aktor petugas operator memiliki 4 fungsionalitas yaitu login aplikasi, mengubah profil, menerima laporan kebakaran masuk dan melihat riwayat

laporan kebakaran. Petugas operator tidak perlu melakukan daftar karena akan didaftarkan melalui admin. Aplikasi yang digunakan petugas operator akan berjalan di device android agar lebih mudah.

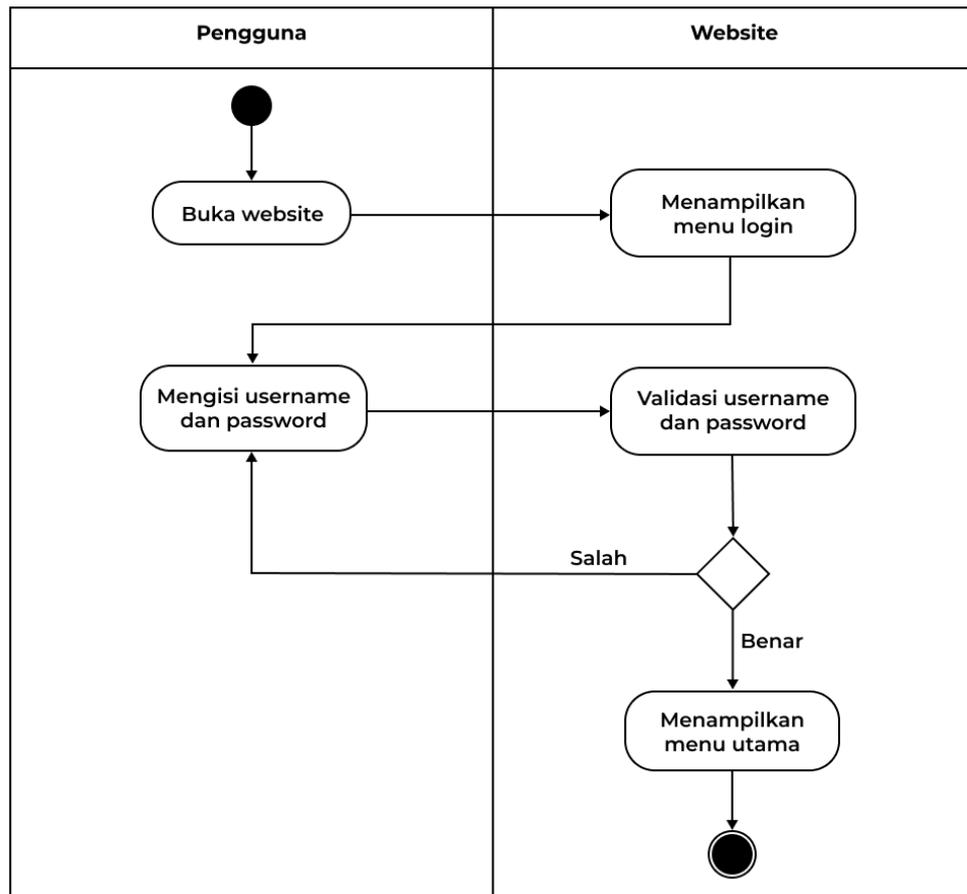
3. Masyarakat memiliki 6 fungsionalitas yaitu daftar, login, mengubah profil, membuat laporan kebakaran, melihat riwayat kebakaran, melihat tentang aplikasi. Semua aktivitas dari aktor tersebut memerlukan proses autentikasi login.

#### **1.4.6 Activity Diagram**

Setelah mengidentifikasi aktor yang berperan didalam sistem, tahap selanjutnya adalah mengidentifikasi proses bisnis yang akan dijalankan menggunakan *activity diagram*. *Activity Diagram* merupakan diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem dari proses status awal sampai status akhir. Berikut merupakan *activity diagram* yang ada dalam aplikasi android dan website yang akan dikembangkan.

##### **a. Activity Diagram Proses Login Admin**

Gambar dibawah ini menunjukkan alur dari rangkaian aktivitas pada proses login admin.

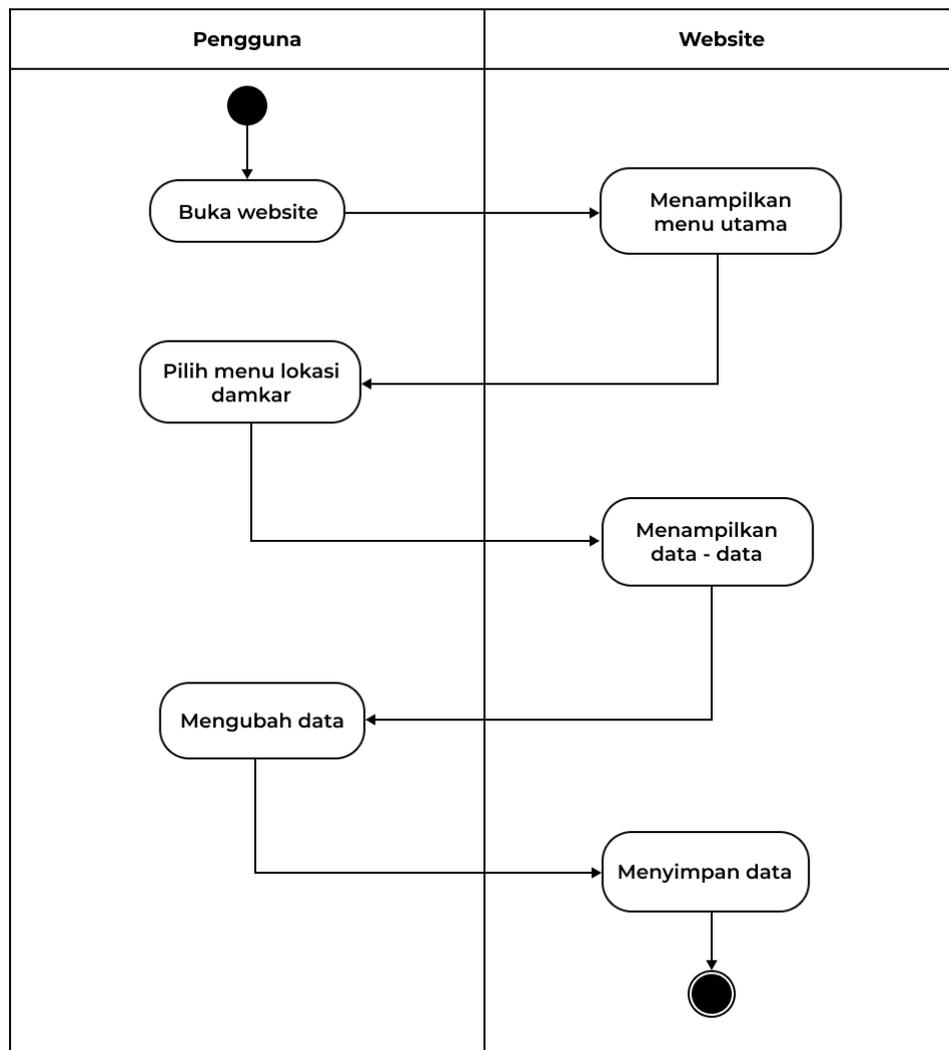


**Gambar 3.9** Activity Diagram Login Admin

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana proses login admin. Dimulai dari admin membuka alamat website kemudian akan ditampilkan form login yang harus diisi oleh admin berupa username dan password. Lalu sistem akan memvalidasi apakah username dan password yang dimasukan cocok, jika cocok maka admin akan diarahkan ke dashboard admin. Jika salah maka akan dikembalikan kepada pengisian form login.

**b. Activity Diagram Proses Admin Kelola Lokasi Pemadam**

Gambar dibawah ini menunjukkan alur dari rangkaian aktivitas pada proses kelola lokasi pemadam.

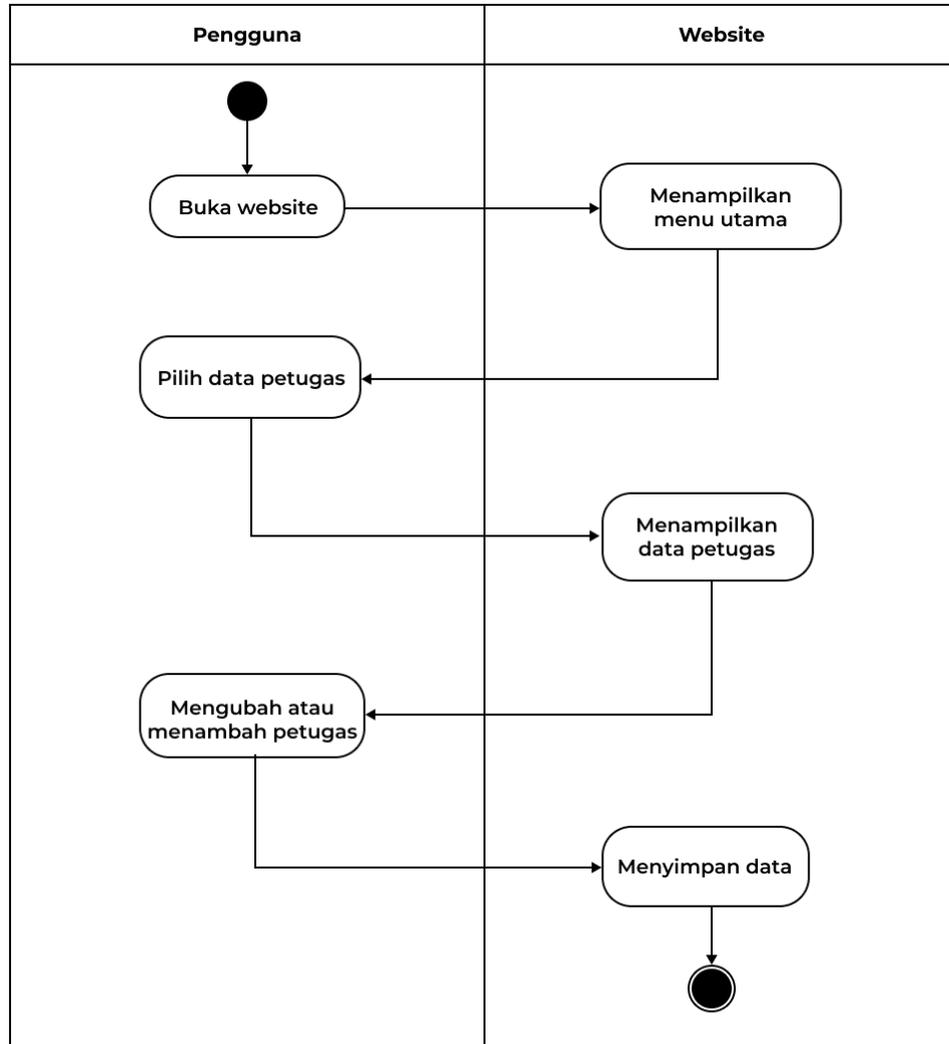


**Gambar 3.10** Activity Diagram Admin Kelola Lokasi Pemadam

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana admin mengelola lokasi pemadam kebakaran. Kelola lokasi damkar hanya dapat dilakukan oleh admin melalui website yang terhubung dengan database. Dimulai dari admin membuka website kemudian sistem menampilkan menu utama. Selanjutnya admin memilih menu lokasi damkar dan sistem akan menampilkan data-data lokasi damkar. Selanjutnya admin dapat menambah dan mengubah data-data lokasi damkar. Pada menu lokasi damkar berisikan lokasi-lokasi pemadam kebakaran yang ada di Bandar Lampung yang sudah diinputkan oleh admin seperti alamat dan titik koordinat. Selanjutnya sistem akan menyimpan data-data tersebut kedalam database.

c. **Activity Diagram Proses Admin Kelola Data Petugas Operator**

Gambar dibawah ini menunjukkan alur dari rangkaian aktivitas pada proses kelola data petugas operator pemadam kebakaran.

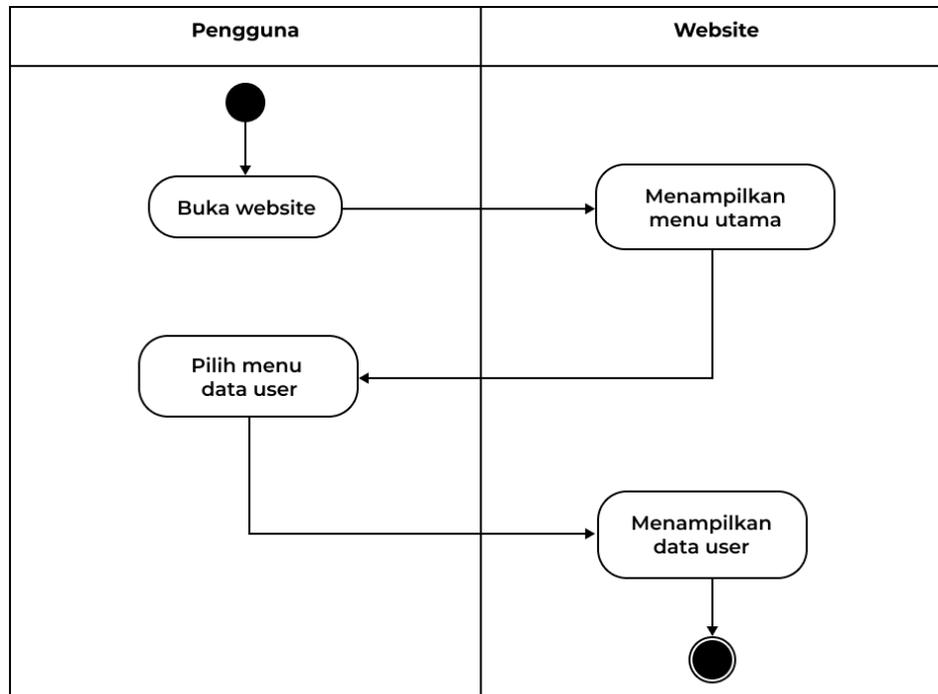


**Gambar 3.11** Activity Diagram Admin Kelola Data Operator

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana admin mengelola data petugas operator pemadam kebakaran. Dimulai dari admin membuka website kemudian sistem akan menampilkan menu utama. Selanjutnya admin memilih menu data petugas operator dan sistem akan menampilkan data petugas operator. Setelah itu admin dapat menambahkan data petugas operator baru, menghapus dan mengubah data kemudian sistem akan memperbarui dan menyimpan data di database. Data petugas operator akan terhubung dengan data lokasi sesuai dimana petugas ditempatkan.

**d. Activity Diagram Proses Admin Kelola Data User**

Gambar dibawah ini menunjukkan alur dari rangkaian aktivitas pada proses kelola data user.

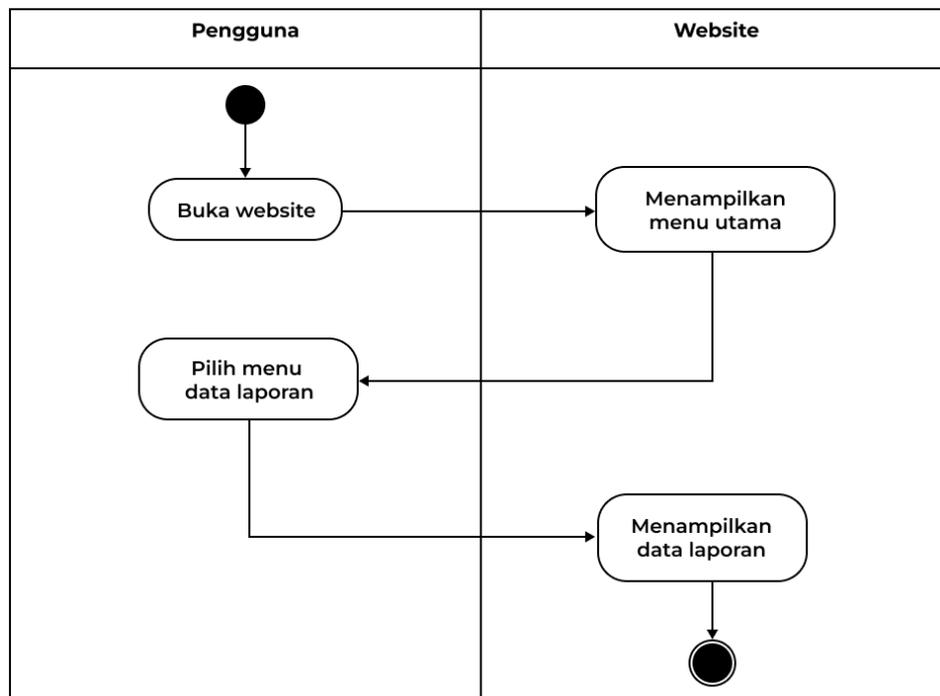


**Gambar 3.12** Activity Diagram Admin Kelola Data User

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana admin mengelola data user masyarakat. Dimulai dari admin membuka website, kemudian sistem akan menampilkan menu utama. Selanjutnya admin memilih menu data user dan sistem akan menampilkan data user secara lengkap. Admin dapat melihat siapa saja user yang mendaftar dan menggunakan aplikasi tersebut. Admin dapat menambahkan, menghapus, mengubah data pada halaman kelola data user.

**e. Activity Diagram Proses Admin Kelola Data Laporan Kebakaran**

Gambar dibawah ini menunjukkan alur dari rangkaian aktivitas pada proses kelola data laporan kebakaran.

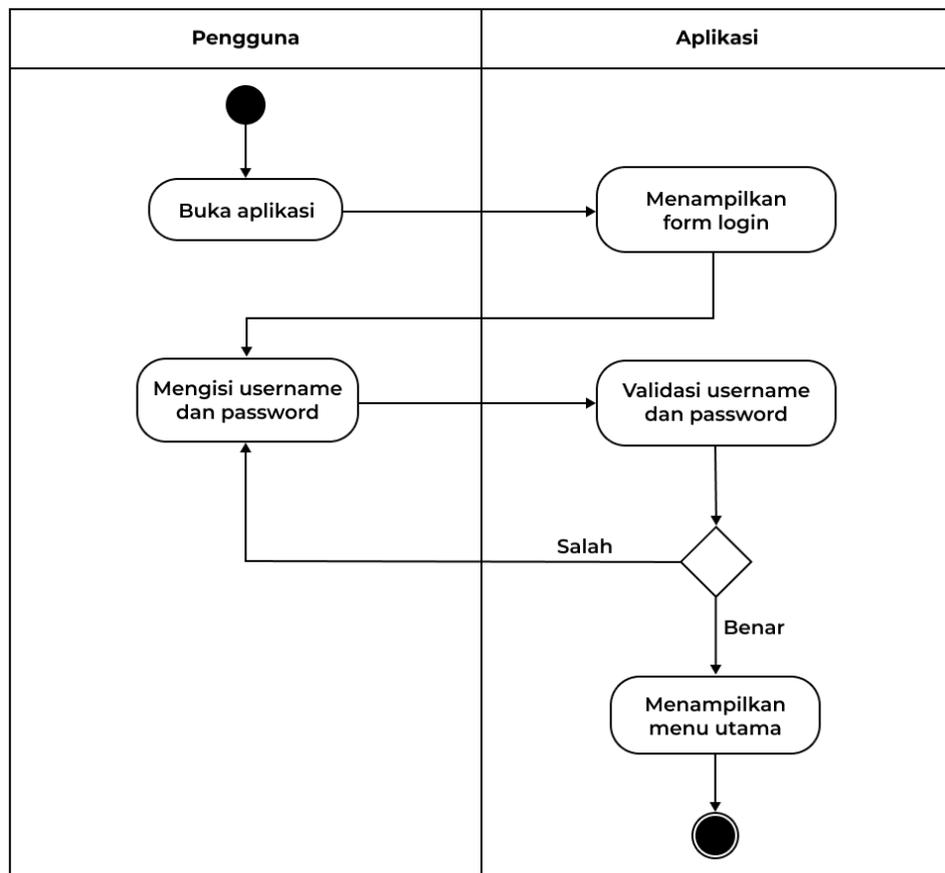


**Gambar 3.13** Activity Diagram Admin Kelola Data Laporan Kebakaran

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana proses admin mengelola kelola data laporan kebakaran. Dimulai dari admin membuka website, kemudian sistem menampilkan menu utama. Selanjutnya admin memilih menu data laporan dan sistem akan menampilkan data laporan. Admin dapat melihat laporan kebakaran yang masuk dan progress dari laporan tersebut apakah sudah selesai ditangani atau belum.

**f. Activity Diagram Proses Login Petugas Operator**

Gambar dibawah ini adalah alur dari rangkaian aktivitas pada proses login petugas operator.

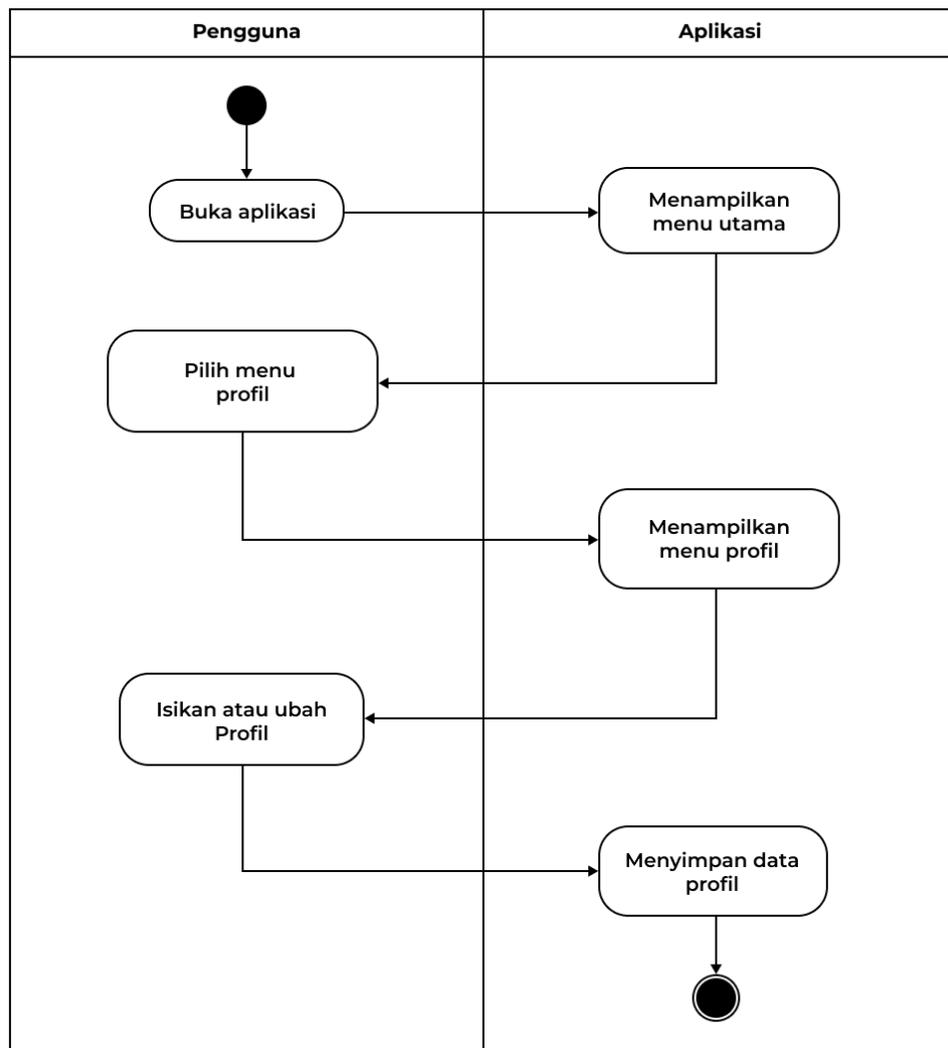


**Gambar 3.14** Activity Diagram Login Operator

Fungsionalitas pada aktivitas diagram proses ini adalah menjelaskan bagaimana proses petugas operator login ke aplikasi. Ketika petugas operator membuka aplikasi maka akan langsung diarahkan ke halaman login untuk mengisi username dan password, kemudian sistem akan memvalidasi apakah username dan password yang dimasukkan sudah benar atau salah. Jika salah maka akan diminta untuk mengisi kembali, namun jika benar maka akan langsung diarahkan ke halaman utama aplikasi. Jika petugas operator belum terdaftar, maka akan didaftarkan oleh admin.

**g. Activity Diagram Proses Profil Petugas Operator**

Gambar dibawah ini adalah alur dari rangkaian aktivitas pada profil petugas operator.

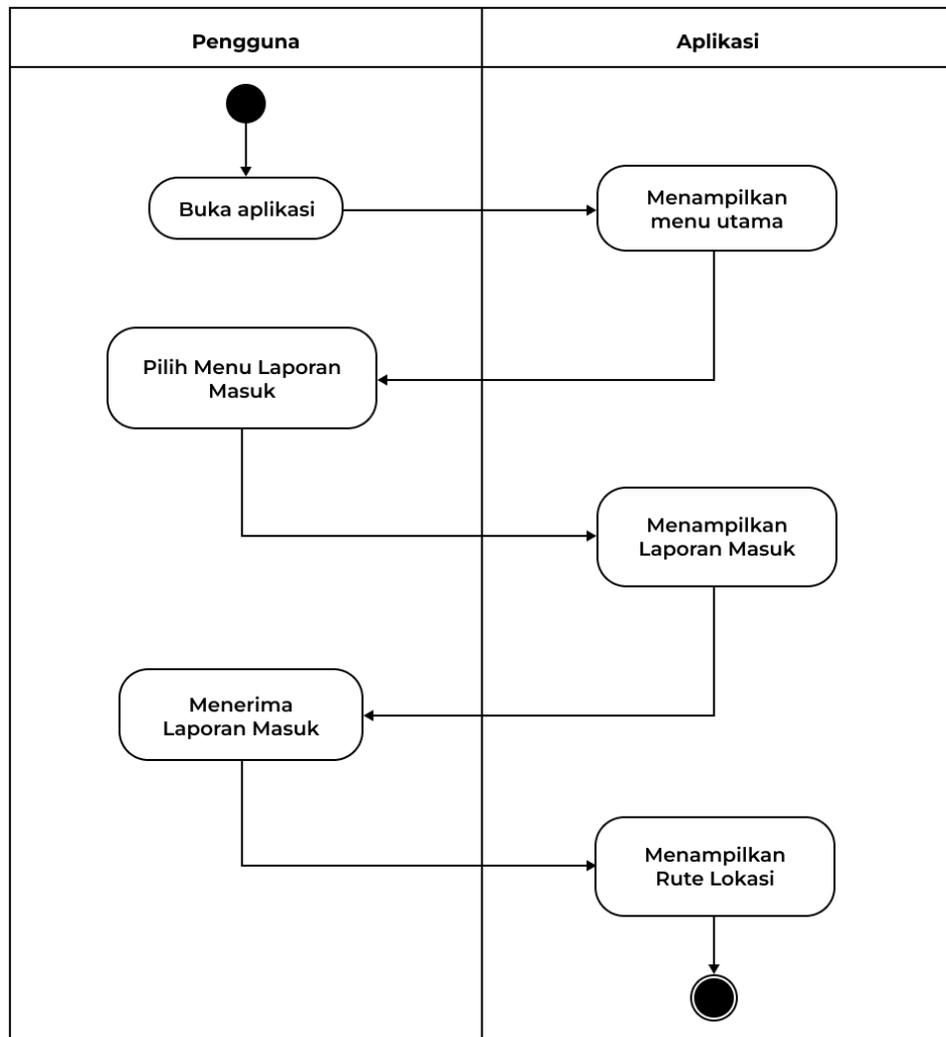


**Gambar 3.15** Activity Diagram Profil Operator

Fungsionalitas pada aktivitas diagram proses profil petugas operator adalah menjelaskan bagaimana petugas dapat melihat dan mengubah profilnya. Dimulai dari petugas operator membuka aplikasi, kemudian sistem akan menampilkan menu utama. Selanjutnya petugas operator memilih menu profil dan sistem akan menampilkan halaman profil yang berisi data petugas. Setelah itu petugas operator dapat mengubah data atau hanya melihat, kemudian sistem akan memperbarui dan menyimpan data ke database

#### **h. Activity Diagram Proses Petugas Operator Menerima Laporan Kebakaran**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses petugas operator menerima laporan kebakaran.

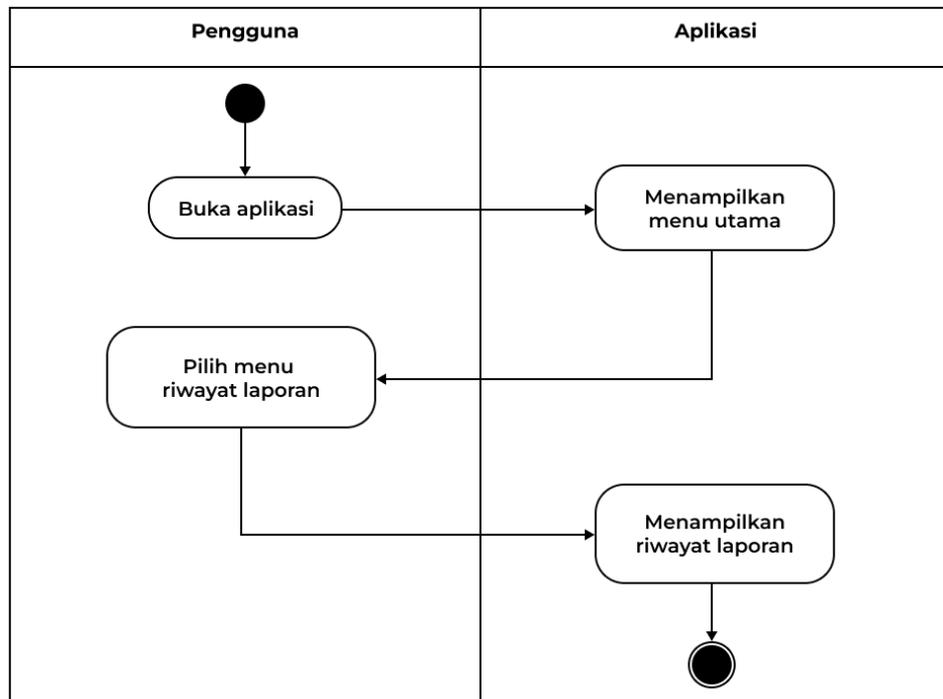


**Gambar 3.16** Activity Diagram Operator Menerima Laporan Kebakaran

Fungsionalitas pada aktivitas diagram proses ini adalah menjelaskan bagaimana proses petugas operator menerima laporan kebakaran. Dimulai dari petugas operator membuka aplikasi, kemudian sistem menampilkan menu utama. Selanjutnya petugas operator memilih menu laporan masuk dan sistem akan menampilkan halaman laporan masuk. Selanjutnya petugas operator menerima laporan masuk dan sistem akan menampilkan rute lokasi kebakaran yang menggunakan API Google Maps.

**i. Activity Diagram Proses Petugas Operator Riwayat Laporan**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses petugas operator melihat riwayat laporan.

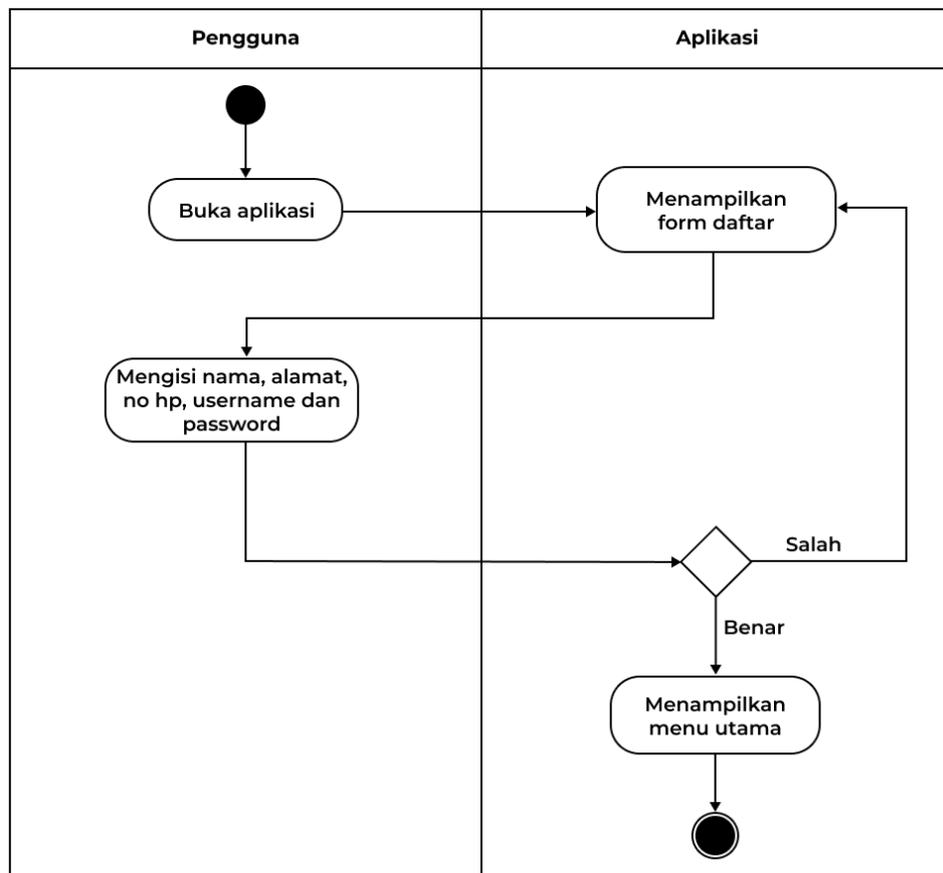


**Gambar 3.17** Activity Diagram Operator Kelola Riwayat Laporan

Fungsionalitas pada aktivitas diagram proses ini adalah menjelaskan bagaimana proses petugas operator melihat riwayat laporan kebakaran. Dimulai dari membuka aplikasi kemudian sistem menampilkan menu utama. Selanjutnya petugas operator memilih menu riwayat laporan dan sistem akan menampilkan halaman riwayat laporan.

#### **j. Activity Diagram Proses Register Masyarakat**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses register pada masyarakat.

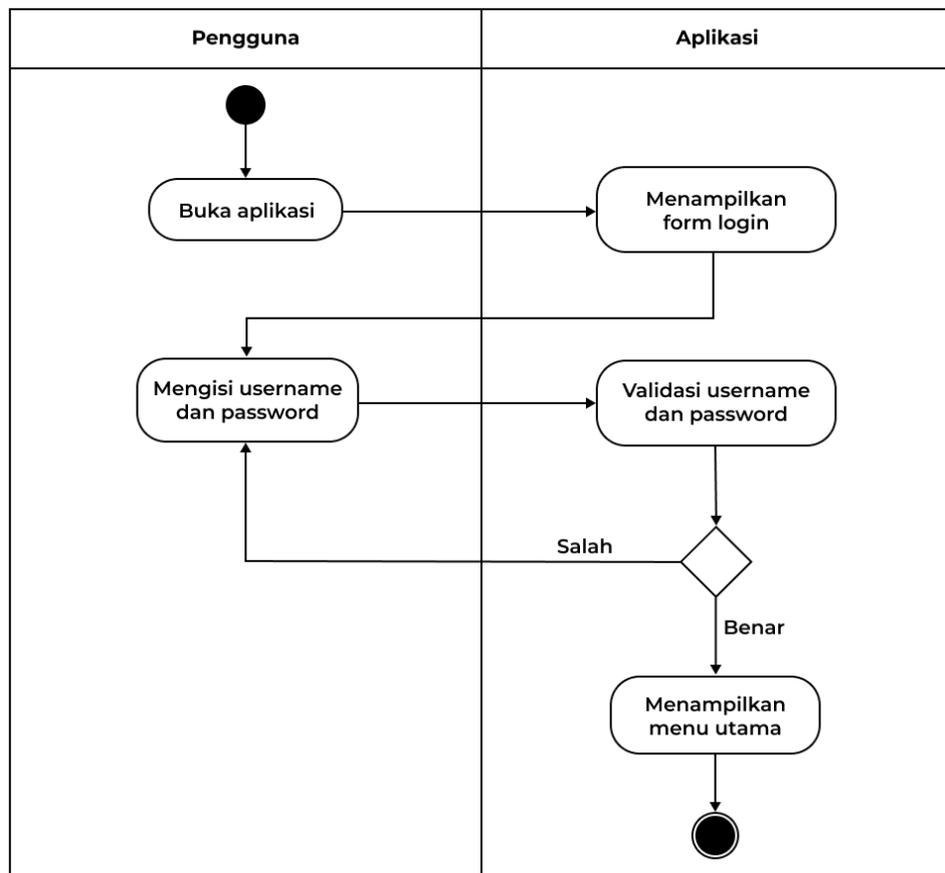


**Gambar 3.18** Activity Diagram Register User

Fungsionalitas pada aktivitas diagram proses ini adalah menjelaskan bagaimana proses *register* akun user sebagai masyarakat. Dimulai dari membuka aplikasi, kemudian sistem akan menampilkan form daftar lalu user mengisi nama, alamat, no hp, *username* dan *password*. Jika data-data yang diisi salah, maka sistem akan mengembalikan ke halaman form. Jika data-data yang diisi sudah benar, maka sistem akan menampilkan halaman menu utama.

#### **k. Activity Diagram Proses Login Masyarakat**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses *login* pada user sebagai masyarakat.

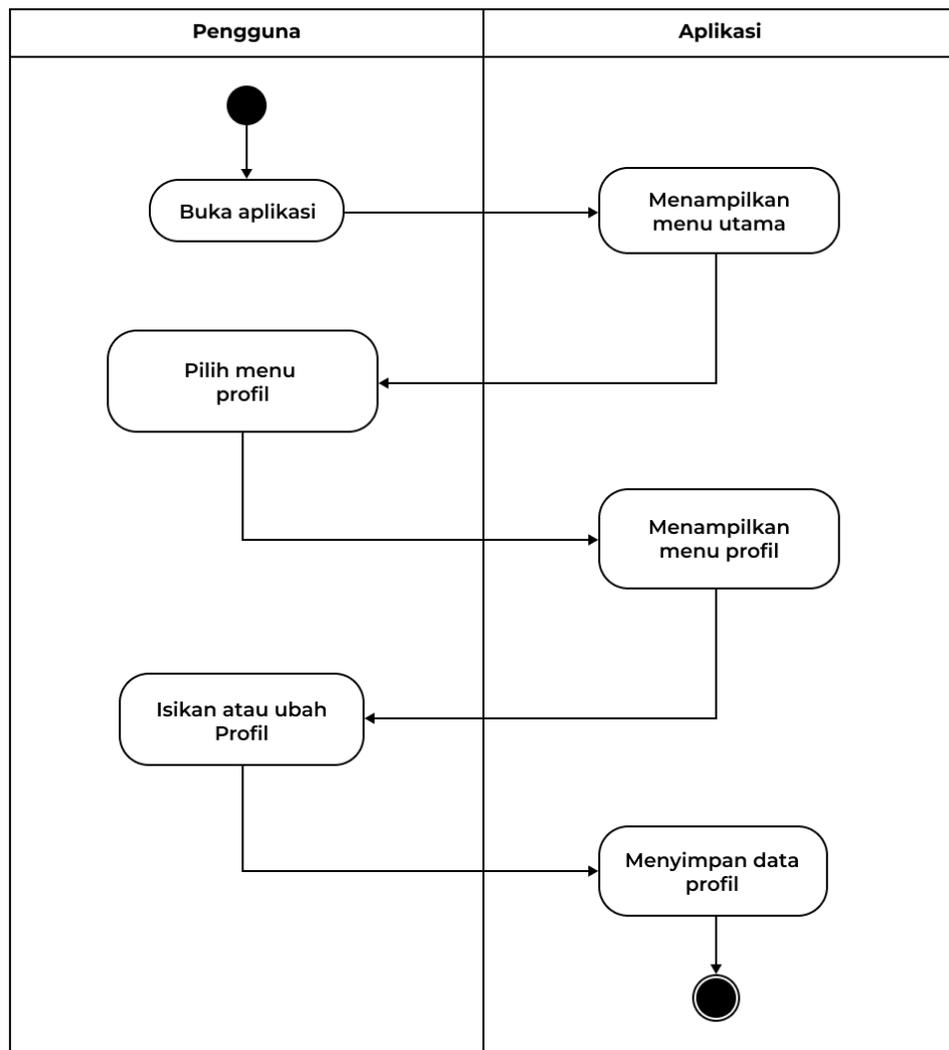


**Gambar 3.19** Activity Diagram Login user

Fungsionalitas pada aktivitas diagram proses login pada masyarakat adalah menjelaskan bagaimana proses login user ke aplikasi. Dimulai dari user membuka aplikasi, kemudian sistem akan menampilkan form login. Selanjutnya user mengisi username dan password yang telah terdaftar di database. Kemudian sistem akan memvalidasi username dan password apakah yang dimasukkan sudah benar atau salah. Jika salah, maka sistem akan mengembalikan ke halaman form login dan user diminta untuk memasukkan lagi username dan password. Jika benar, maka sistem akan menampilkan halaman menu utama. Tapi jika user belum memiliki username dan password, maka akan diarahkan ke halaman register.

#### **I. Activity Diagram Proses Profil Masyarakat**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses profile pada user sebagai masyarakat.

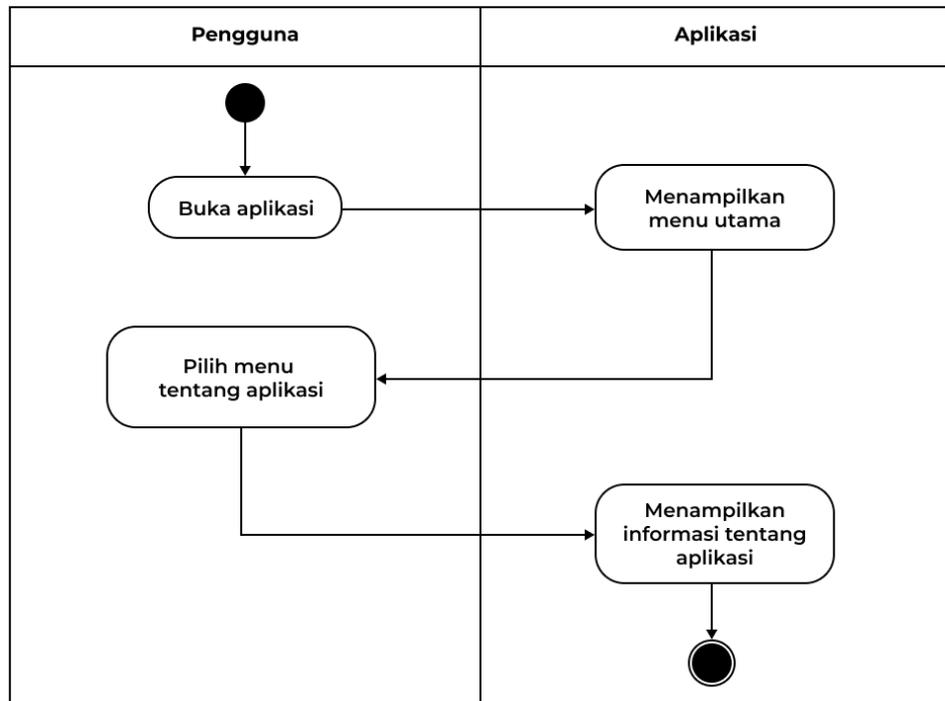


**Gambar 3.20** Activity Diagram Profil *User*

Fungsionalitas pada diagram aktivitas ini adalah menjelaskan bagaimana proses user melihat dan mengubah profilnya. Dimulai dari user membuka aplikasi, kemudian sistem akan menampilkan menu utama. Selanjutnya user memilih menu profile dan sistem akan menampilkan halaman profile. Kemudian user dapat mengubah data-data seperti nama, username, password, alamat dan no hp. Setelah itu sistem akan menyimpan data ke database.

#### **m. Activity Diagram Proses Tentang Aplikasi**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses tentang aplikasi pada user sebagai masyarakat.

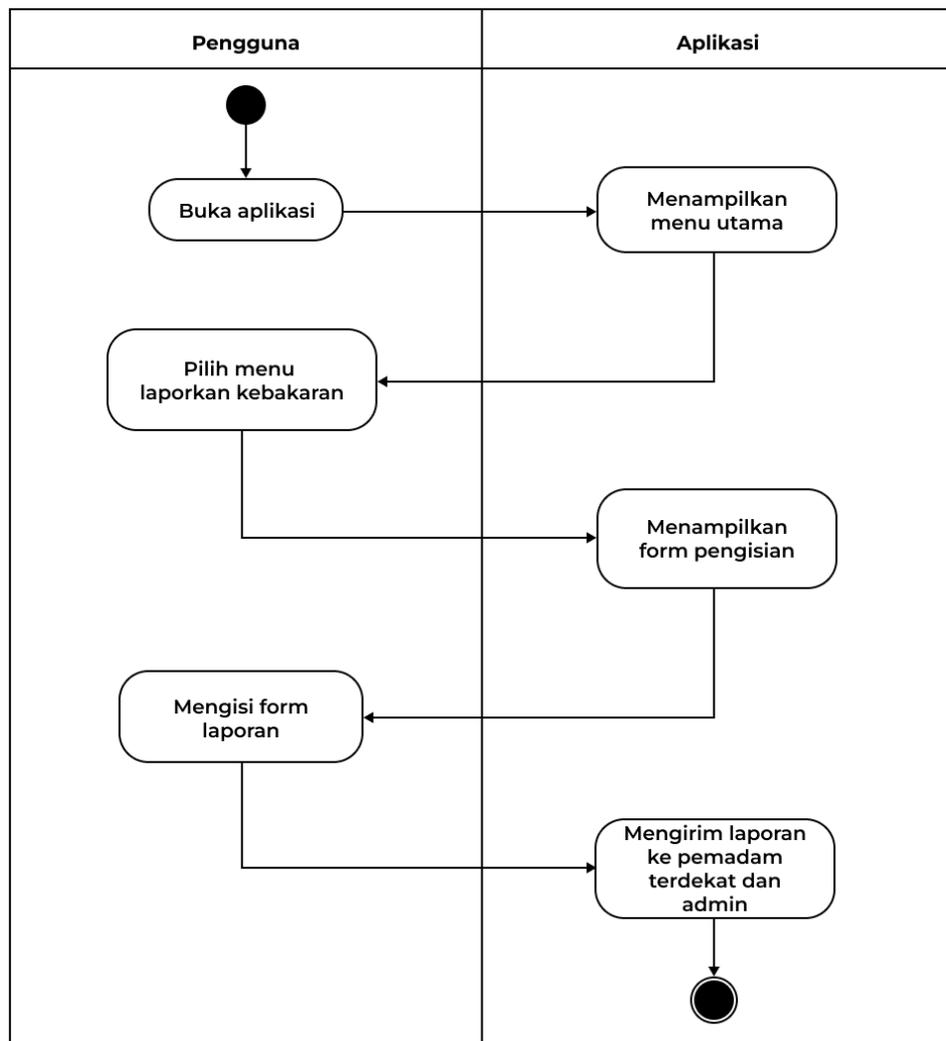


**Gambar 3.21** Activity Diagram Tentang Aplikasi

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana user melihat halaman tentang aplikasi. Dimulai dari user membuka aplikasi, kemudian sistem menampilkan menu utama. Selanjutnya user memilih menu tentang aplikasi dan sistem akan menampilkan halaman yang berisikan tentang aplikasi tersebut.

**n. Activity Diagram Proses Laporkan Kebakaran**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses melaporkan kebakaran.

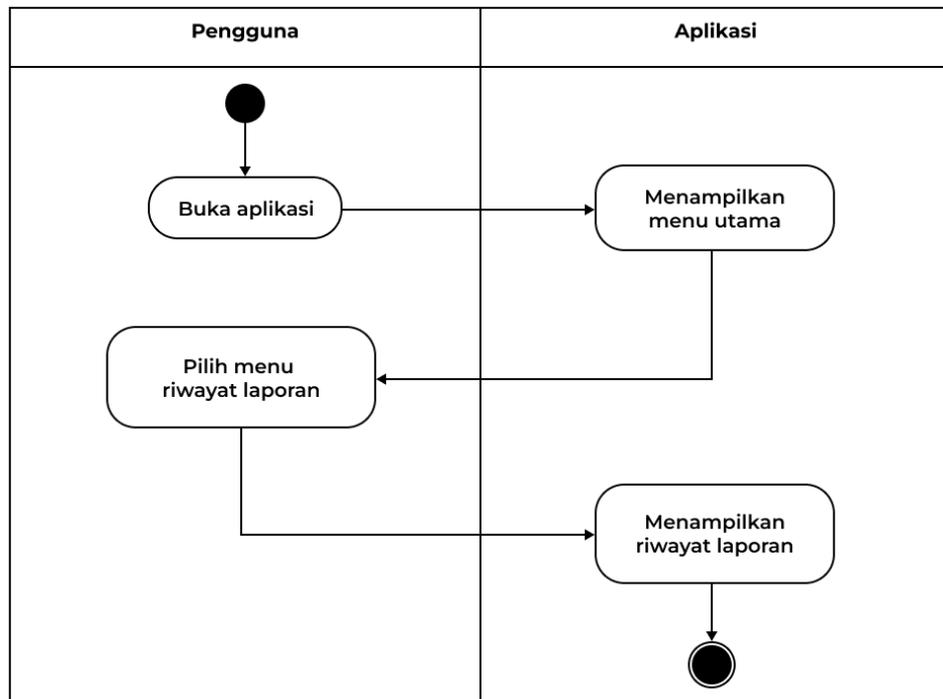


**Gambar 3.22** Activity Diagram Laporkan Kebakaran

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana porses masyarakat melaporkan terjadinya kebakaran. Dimulai dari user membuka aplikasi kemudian sistem menampilkan menu utama. Selanjutnya user memilih menu laporkan kebakaran dan sistem akan menampilkan form pengisian. Selanjutnya user mengisi form laporan seperti nama, waktu, lokasi dan foto. Untuk nama, waktu dan lokasi akan terisi otomatis sesuai dengan nama di profil, waktu melaporkan dan lokasi terkini terjadinya kebakaran. Setelah mengisi form laporan, maka sistem akan mengirimkan laporan kepada petugas operator pemadam kebakaran terdekat dan kepada admin untuk diproses.

**o. Activity Diagram Proses Riwayat Laporan Kebakaran**

Gambar dibawah ini adalah alur dari rangkaian aktivitas proses melihat riwayat laporan kebakaran.

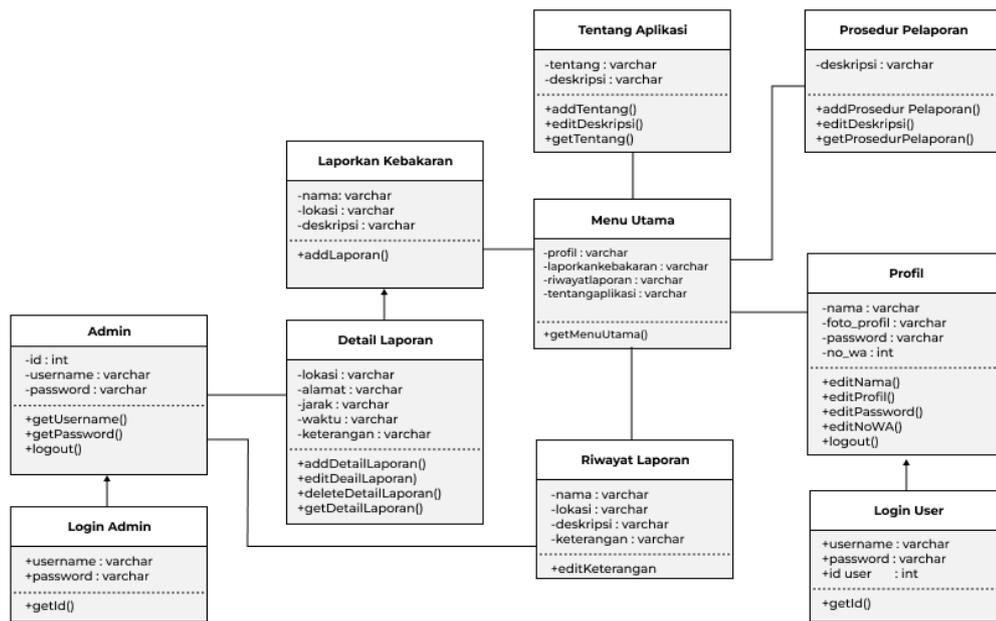


**Gambar 3.23** *Activity Diagram* Riwayat Laporan Kebakaran

Fungsionalitas pada aktivitas diagram ini adalah menjelaskan bagaimana user melihat laporan kebakaran yang telah dikirim. Dimulai dari user membuka aplikasi kemudian sistem menampilkan halaman menu utama. Selanjutnya user memilih menu riwayat laporan dan sistem akan menampilkan halaman riwayat laporan.

#### 1.4.7 *Class Diagram*

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain. Class diagram pada aplikasi Smart Service Pelayanan Kebakaran dapat dilihat pada gambar 3.24.



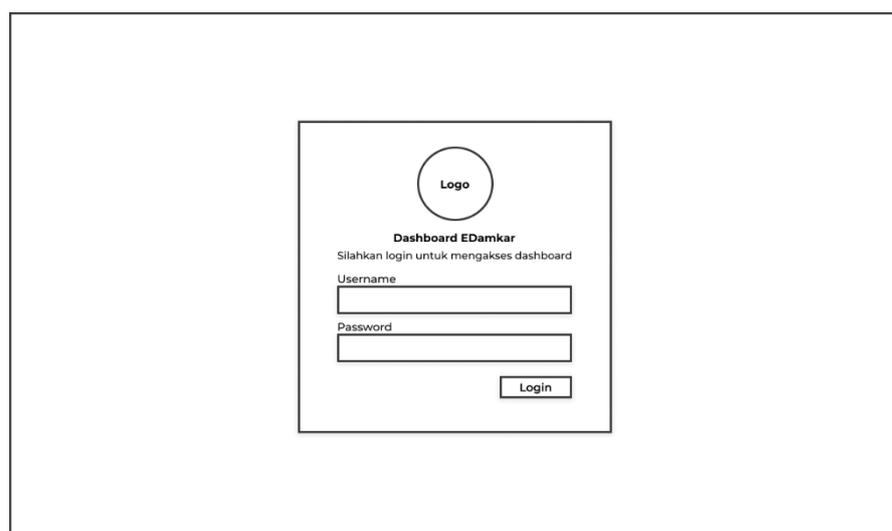
**Gambar 3.24** Class Diagram

### 1.4.8 Desain Antarmuka Aplikasi

Desain aplikasi merupakan rancangan bagaimana bentuk dan rupa tampilan aplikasi yang akan dibuat. Berikut adalah rancangan desain tampilan antarmuka:

a. Rancangan Tampilan Halaman Login Admin

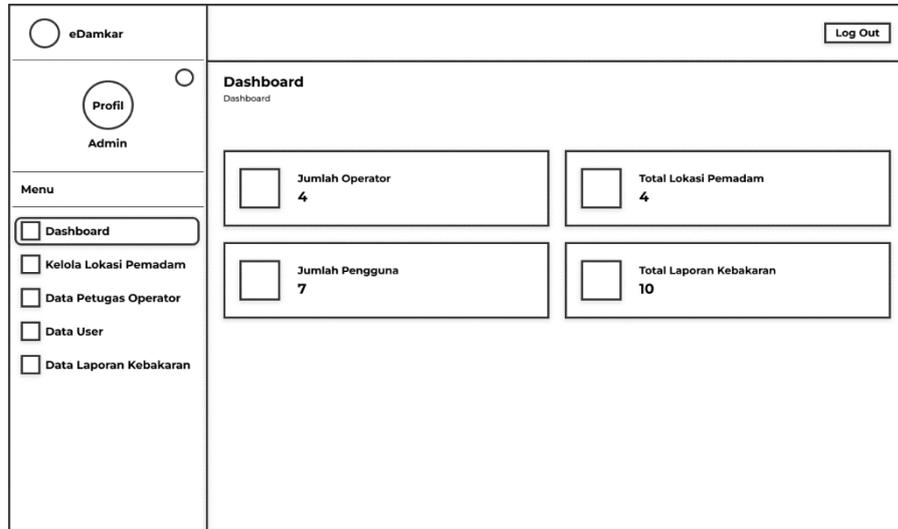
Pada rancangan tampilan login admin menampilkan form login admin melalui website yang berisi username dan password.



**Gambar 3.25** Rancangan Tampilan Login Admin

b. Rancangan Tampilan Halaman Dashboard Admin

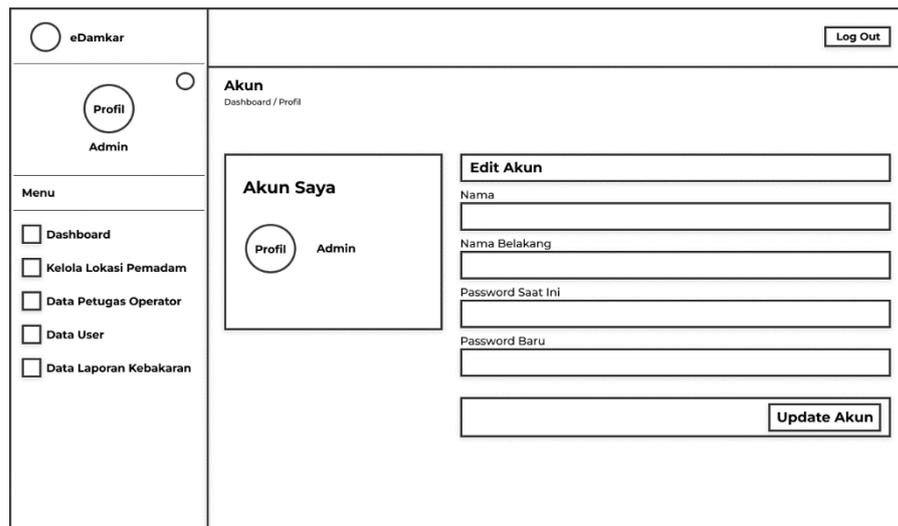
Pada rancangan tampilan dashboard admin menampilkan menu dashboard admin melalui website yang berisi jumlah operator, total pos pemadam, jumlah pengguna dan jumlah laporan kebakaran.



**Gambar 3.26** Rancangan Tampilan Dashboard Admin

c. Rancangan Tampilan Halaman Profil Admin

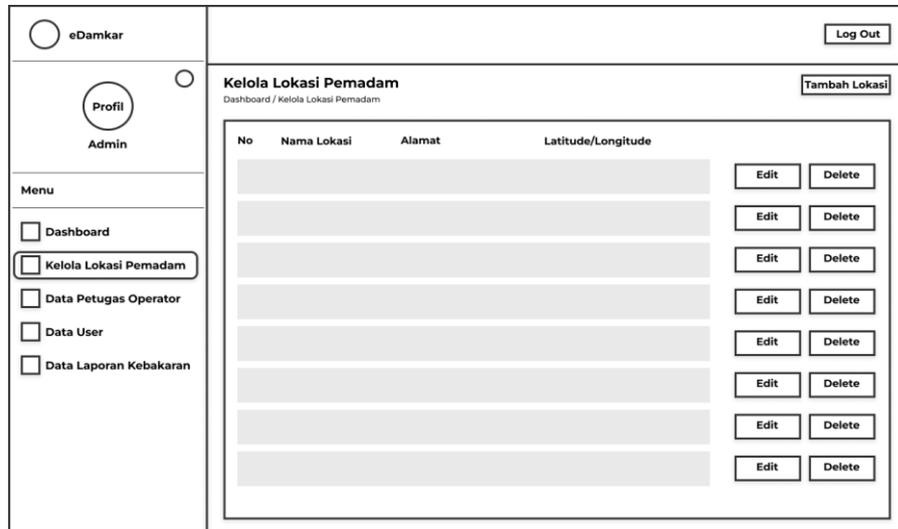
Pada rancangan tampilan profil admin berupa website, berisi tentang data admin dan admin dapat mengubah data tersebut.



**Gambar 3.27** Rancangan Tampilan Profil Admin

d. Rancangan Tampilan Halaman Admin Kelola Lokasi Pemadam

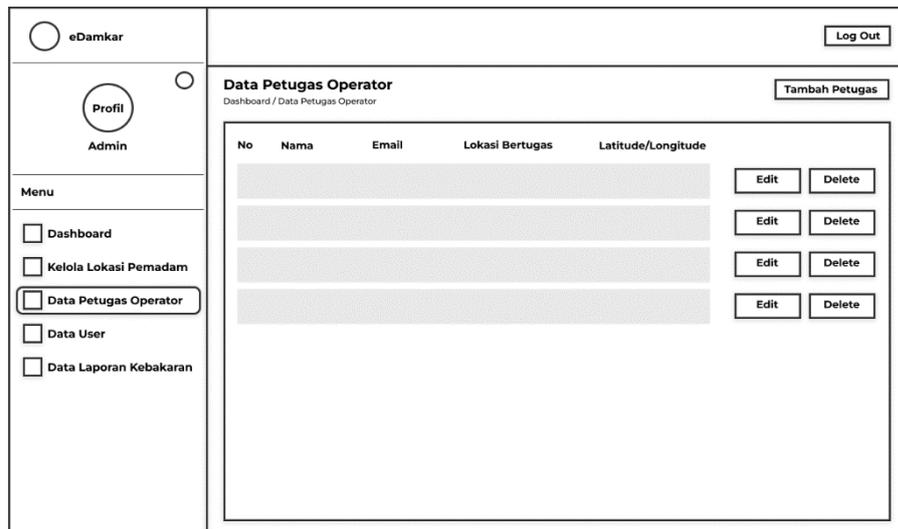
Pada rancangan tampilan halaman admin kelola lokasi pemadam berupa website. Admin dapat menambah, menghapus dan mengubah data mengenai lokasi pemadam kebakaran.



**Gambar 3.28** Rancangan Tampilan Admin Kelola Lokasi Pemadam

e. Rancangan Tampilan Halaman Admin Kelola Operator

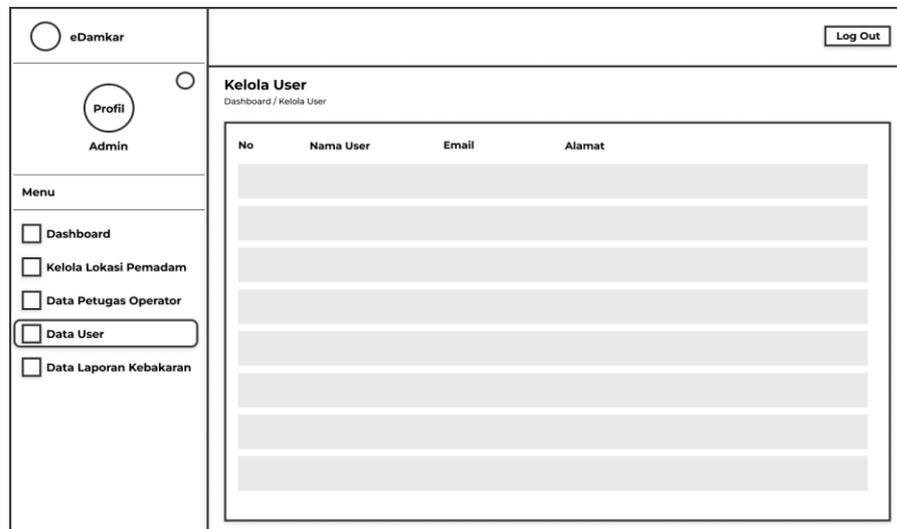
Pada rancangan tampilan admin kelola operator berupa website. Admin dapat menambah, menghapus dan mengubah data mengenai operator.



**Gambar 3.29** Rancangan Tampilan Admin Kelola Opeartor

f. Rancangan Tampilan Halaman Admin Data User

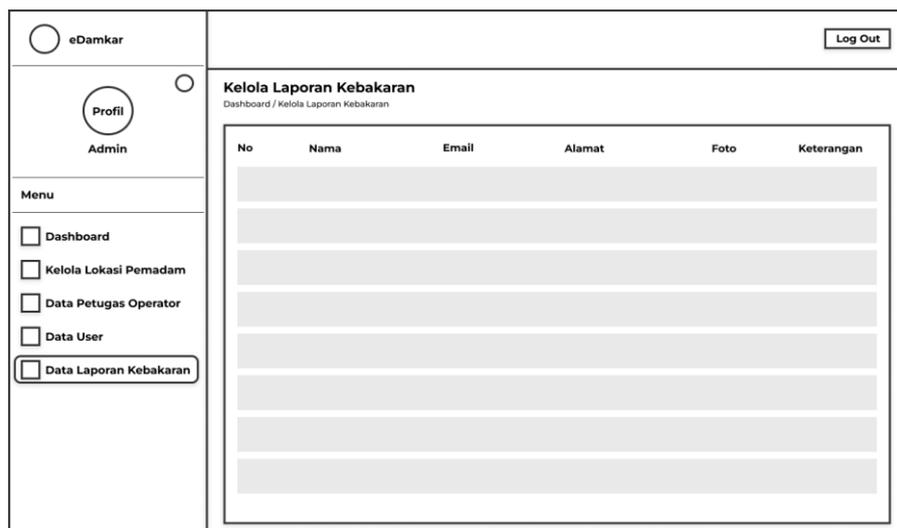
Pada rancangan tampilan halaman admin data user berupa website. Admin dapat melihat siapa saja yang sudah mendaftar aplikasi smart service layanan pemadam kebakaran.



**Gambar 3.30** Rancangan Tampilan Admin Kelola Data User

g. Rancangan Tampilan Halaman Admin Data Laporan Kebakaran

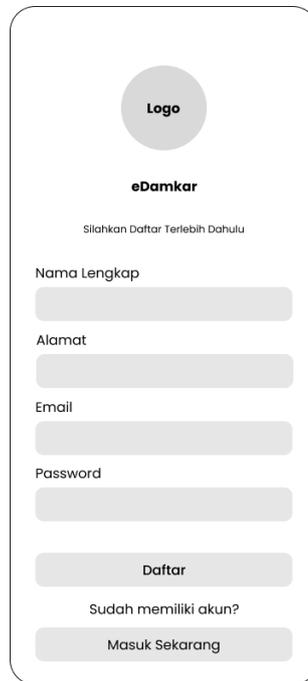
Pada rancangan tampilan halaman admin data laporan kebakaran berupa website. Admin dapat melihat laporan kebakaran yang dilaporkan oleh masyarakat.



**Gambar 3.31** Rancangan Tampilan Admin Kelola Data Laporan

h. Rancangan Tampilan Halaman Daftar User

Pada rancangan ini berupa aplikasi android yang menampilkan form daftar yang digunakan untuk mengisi data ketika user mendaftar.

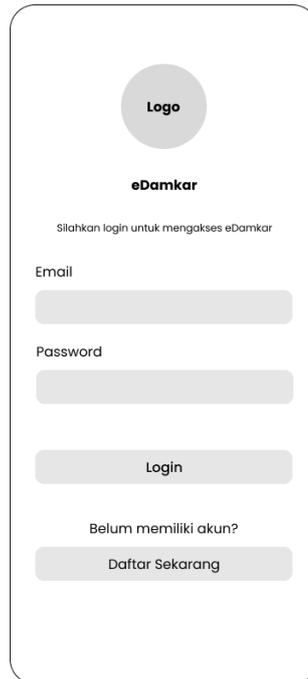


The image shows a mobile application registration screen. At the top, there is a circular logo placeholder labeled "Logo". Below it, the text "eDankar" is displayed. A message "Silahkan Daftar Terlebih Dahulu" (Please register first) is shown. The form contains four input fields: "Nama Lengkap" (Full Name), "Alamat" (Address), "Email", and "Password". Below the input fields, there is a "Daftar" (Register) button. At the bottom, there is a link "Sudah memiliki akun?" (Already have an account?) with a "Masuk Sekarang" (Login Now) button.

**Gambar 3.32** Rancangan Tampilan *Register User*

i. Rancangan Tampilan Halaman Login User

Pada rancangan ini berupa aplikasi android yang menampilkan form login yang digunakan user untuk masuk ke aplikasi dengan menginput username dan password yang sudah terdaftar pada database.



**Gambar 3.33** Rancangan Tampilan *Login User*

j. Rancangan Tampilan Halaman Menu Utama User

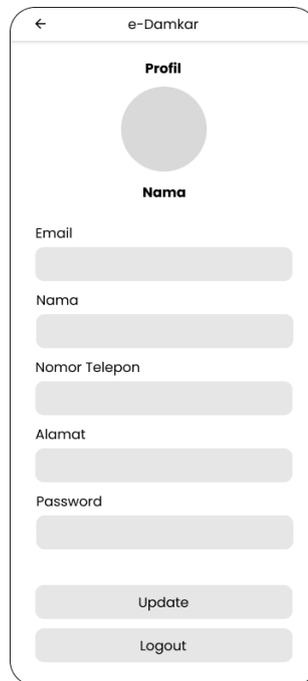
Pada rancangan ini berupa aplikasi android yang menampilkan menu utama seperti profil, laporkan kebakaran, riwayat laporan, prosedur pelaporan, dan tentang aplikasi.



**Gambar 3.34** Rancangan Tampilan Menu Utama User

k. Rancangan Tampilan Halaman Profil User

Pada rancangan ini berupa aplikasi android yang menampilkan data profil user. User dapat mengubah data profile sesuai dengan kebutuhan.



The image shows a mobile application screen for a user profile. At the top, there is a back arrow and the text "e-Damkar". Below this, the word "Profil" is centered. A large grey circle represents a profile picture. Underneath the circle, the word "Nama" is centered. The form consists of several input fields: "Email", "Nama", "Nomor Telepon", "Alamat", and "Password". Each field is represented by a grey rounded rectangle. At the bottom of the form, there are two buttons: "Update" and "Logout", also represented by grey rounded rectangles.

**Gambar 3.35** Rancangan Tampilan Profil User

l. Rancangan Tampilan Halaman Laporkan Kebakaran

Pada rancangan ini berupa aplikasi android yang menampilkan form laporan kebakaran yang harus diisi oleh user ketika melaporkan adanya kejadian kebakaran.

e-Damkar

**Laporan kebakaran**

Maps

Latitude -538955 Longitude 105.2513

Nama  
Otomatis terisi sesuai di profil

No HP  
Otomatis terisi sesuai di profil

Tanggal & Waktu  
Otomatis terisi

Foto

Kirim

**Gambar 3.36** Rancangan Tampilan Laporan Kebakaran

m. Rancangan Tampilan Halaman Pemberitahuan Laporan Terkirim

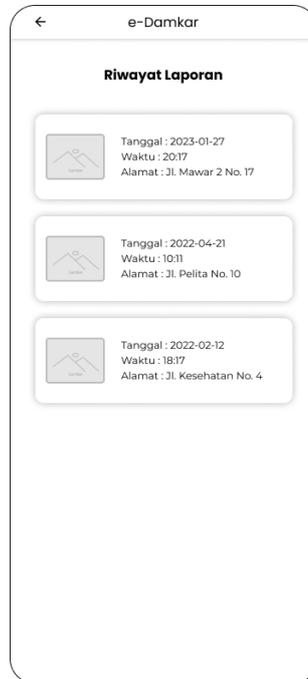
Pada rancangan tampilan ini berupa aplikasi android yang menampilkan pemberitahuan bahwa laporan yang di kirim user telah sampai ke operator pemadam terdekat.



**Gambar 3.37** Rancangan Tampilan Pemberitahuan Laporan Terkirim

n. Rancangan Tampilan Halaman Riwayat Laporan

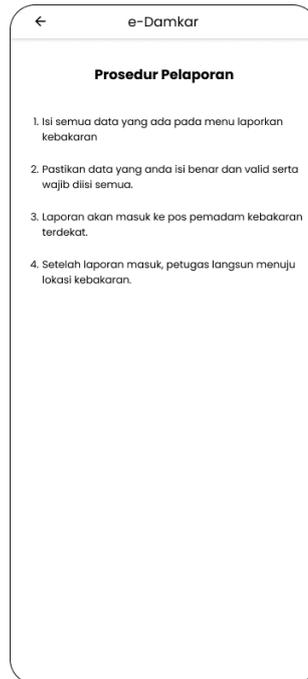
Pada rancangan tampilan ini berupa aplikasi android yang menampilkan data riwayat laporan kebakaran yang pernah dilaporkan oleh user.



**Gambar 3.38** Rancangan Tampilan Riwayat Laporan

o. Rancangan Tampilan Halaman Prosedur Pelaporan

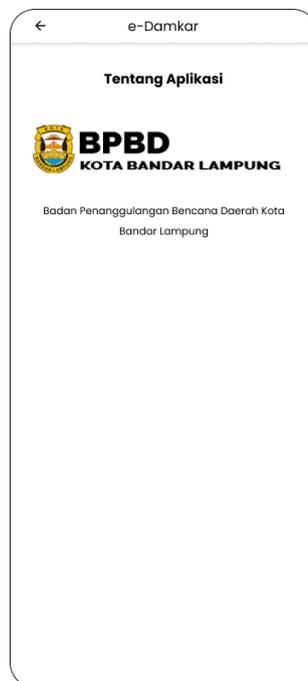
Pada rancangan tampilan ini berupa aplikasi android yang menampilkan bagaimana cara melaporkan kebakaran melalui aplikasi tersebut.



**Gambar 3.39** Rancangan Tampilan Prosedur Pelaporan

p. Rancangan Tampilan Halaman Tentang Aplikasi

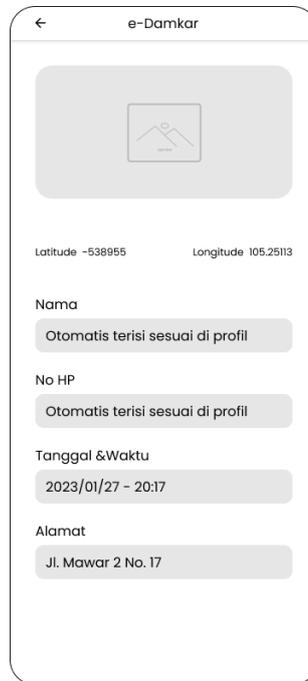
Pada rancangan tampilan ini berupa aplikasi android yang menampilkan informasi tentang aplikasi.



**Gambar 3.40** Rancangan Tampilan Tentang Aplikasi

q. Rancangan Tampilan Halaman Detail Laporan

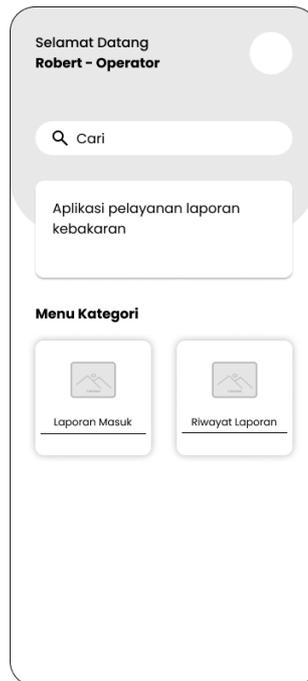
Pada rancangan tampilan ini berupa aplikasi android yang menampilkan detail laporan yang ada di riwayat laporan.



**Gambar 3.41** Rancangan Tampilan Detail Laporan

r. Rancangan Tampilan Halaman Menu Utama Operator

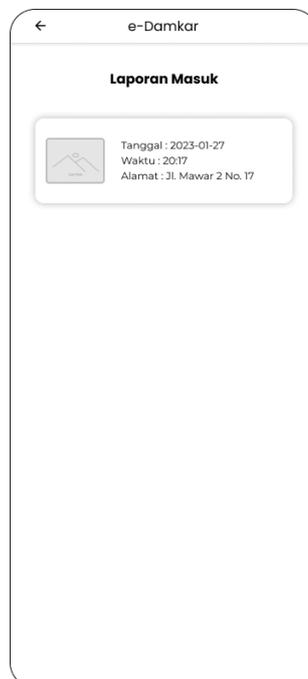
Pada rancangan tampilan halaman menu utama operator berupa aplikasi android yang menampilkan menu seperti laporan masuk dan riwayat laporan.



**Gambar 3.42** Rancangan Tampilan Menu Utama Operator

s. Rancangan Tampilan Halaman Laporan Masuk Operator

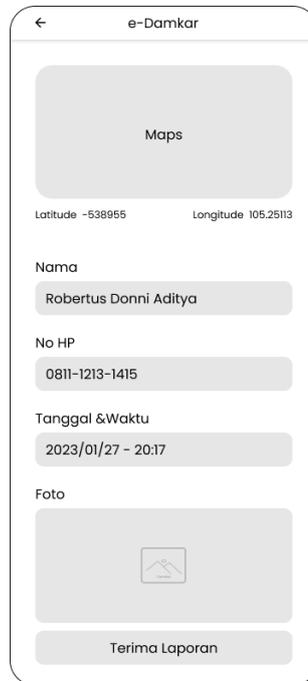
Pada rancangan tampilan halaman ini berupa aplikasi android yang menampilkan laporan masuk yang dikirim oleh masyarakat.



**Gambar 3.43** Rancangan Tampilan Laporan Masuk

t. Rancangan Tampilan Halaman Detail Laporan Masuk

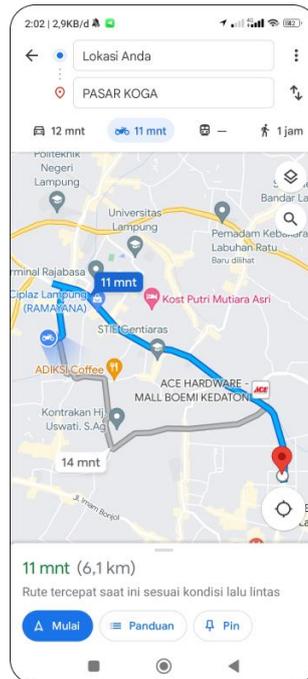
Pada rancangan tampilan halaman ini berupa aplikasi android yang menampilkan detail laporan masuk seperti nama, no hp, alamat, lokasi, tanggal, waktu, dan foto. Kemudian terdapat tombol terima laporan yang berfungsi untuk menerima laporan kebakaran.



**Gambar 3.44** Rancangan Tampilan Detail Laporan Masuk

u. Rancangan Tampilan Halaman Rute

Pada rancangan tampilan halaman rute menampilkan rute terdekat dari pos pemadam ke lokasi kebakaran yang diarahkan ke aplikasi Google Maps.



**Gambar 3.45** Rancangan Tampilan Rute Perjalanan

## 1.5 Sprint

Tahap ini dibagi dalam 2 sesi yaitu *sprint panning* dan *sprint backlog*. Pada *sprint planning*, peneliti melakukan evaluasi daftar yang ada pada *product backlog*. Peneliti memikirkan dengan detail setiap fitur yang nantinya akan dibangun dengan estimasi waktu pengerjaan seperti table dibawah ini.

**Tabel 3.6** *Sprint Planning*

| Aktor                | <i>Sprint Planning</i>        | Estimasi (Waktu/Hari) |
|----------------------|-------------------------------|-----------------------|
| Admin                | Login Admin                   | 1                     |
|                      | Dashboard Admin               | 1                     |
|                      | Profil Admin                  | 1                     |
|                      | Data Petugas Operator Pemadam | 1                     |
|                      | Kelola Lokasi Pemadam         | 1                     |
|                      | Data User                     | 1                     |
|                      | Data Laporan                  | 1                     |
| User<br>(Masyarakat) | Register User                 | 1                     |
|                      | Login User                    | 1                     |

|                  |                            |   |
|------------------|----------------------------|---|
|                  | Halaman Menu Utama         | 1 |
|                  | Dashboard dan Profil User  | 1 |
|                  | Halaman Laporkan Kebakaran | 1 |
|                  | Halaman Riwayat Laporan    | 1 |
|                  | Halaman Tentang Aplikasi   | 1 |
| Petugas Operator | Login                      | 1 |
|                  | Halaman Profil             | 1 |
|                  | Halaman Laporan Masuk      | 1 |
|                  | Halaman Riwayat Laporan    | 1 |

Selanjutnya pada *sprint backlog*, daftar fitur yang telah dibuat pada *product backlog* dan *sprint planning* dikumpulkan untuk tahap pemrograman seperti yang terlihat pada tabel dibawah ini. Setelah fitur dalam *sprint planning* selesai dikerjakan, maka fitur lain akan dilanjutkan *sprint* berikutnya.

**Tabel 3.7** *Sprint Backlog Sprint 1 Halaman Register User*

| Sprint 1                           | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|------------------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                                    |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Register (User Masyarakat) | Membuat Database          | 1                    |   |   |   |   |   |   |
|                                    | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                                    | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                                    | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                                    | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.8** *Sprint Backlog Sprint 2 Halaman Login*

| Sprint 2      | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|---------------|---------------------------|----------------------|---|---|---|---|---|---|
|               |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Login | Membuat Database          | 1                    |   |   |   |   |   |   |
|               | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|               | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|               | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|               | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.9** *Sprint Backlog Sprint 3 Halaman Menu Utama*

| Sprint 3 | Task             | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|----------|------------------|----------------------|---|---|---|---|---|---|
|          |                  | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
|          | Membuat Database | 1                    |   |   |   |   |   |   |

|                    |                           |       |   |  |  |  |  |  |
|--------------------|---------------------------|-------|---|--|--|--|--|--|
| Halaman Menu Utama | Tampilan <i>front-end</i> | 2     |   |  |  |  |  |  |
|                    | <i>Coding</i>             |       | 3 |  |  |  |  |  |
|                    | <i>Testing</i>            |       | 1 |  |  |  |  |  |
|                    | Total                     | 7 Jam |   |  |  |  |  |  |

**Tabel 3.10** *Sprint Backlog Sprint 4* Halaman Profil

| Sprint 4       | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|----------------|---------------------------|----------------------|---|---|---|---|---|---|
|                |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Profil | Membuat Database          | 1                    |   |   |   |   |   |   |
|                | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.11** *Sprint Backlog Sprint 5* Halaman Laporkan Kebakaran

| Sprint 5                   | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|----------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                            |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Laporkan Kebakaran | Membuat Database          | 1                    |   |   |   |   |   |   |
|                            | Tampilan <i>front-end</i> | 2                    |   |   |   |   |   |   |
|                            | <i>Coding</i>             |                      | 3 |   |   |   |   |   |
|                            | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                            | Total                     | 7 Jam                |   |   |   |   |   |   |

**Tabel 3.12** *Sprint Backlog Sprint 6* Halaman Riwayat Kebakaran

| Sprint 6                  | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|---------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                           |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Riwayat Kebakaran | Membuat Database          | 1                    |   |   |   |   |   |   |
|                           | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                           | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                           | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                           | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.13** *Sprint Backlog Sprint 7* Halaman Tentang Aplikasi

| Sprint 7        | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-----------------|---------------------------|----------------------|---|---|---|---|---|---|
|                 |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Tentang | Membuat Database          |                      |   |   |   |   |   |   |
|                 | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                 | <i>Coding</i>             |                      | 1 |   |   |   |   |   |
|                 | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                 | Total                     | 3 Jam                |   |   |   |   |   |   |

**Tabel 3.14** *Sprint Backlog Sprint 8* Halaman Login Admin

| Sprint 8            | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|---------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                     |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Login Admin | Membuat Database          | 1                    |   |   |   |   |   |   |
|                     | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                     | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                     | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                     | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.15** *Sprint Backlog Sprint 9* Halaman Dashboard Admin

| Sprint 9                | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                         |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Dashboard Admin | Membuat Database          | 1                    |   |   |   |   |   |   |
|                         | Tampilan <i>front-end</i> | 2                    |   |   |   |   |   |   |
|                         | <i>Coding</i>             |                      | 4 |   |   |   |   |   |
|                         | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                         | Total                     | 8 Jam                |   |   |   |   |   |   |

**Tabel 3.16** *Sprint Backlog Sprint 10* Halaman Profil Admin

| Sprint 10            | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|----------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                      |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Profil Admin | Membuat Database          | 1                    |   |   |   |   |   |   |
|                      | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                      | <i>Coding</i>             |                      | 1 |   |   |   |   |   |
|                      | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                      | Total                     | 4 Jam                |   |   |   |   |   |   |

**Tabel 3.17** *Sprint Backlog Sprint 11* Halaman Kelola Lokasi Pemadam

| Sprint 11                     | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-------------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                               |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Kelola Lokasi Pemadam | Membuat Database          | 1                    |   |   |   |   |   |   |
|                               | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                               | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                               | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                               | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.18** *Sprint Backlog Sprint 12* Halaman Data Petugas Operator

| Sprint 12 | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-----------|---------------------------|----------------------|---|---|---|---|---|---|
|           |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
|           | Membuat Database          | 1                    |   |   |   |   |   |   |
|           | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |

|                                     |                |       |   |  |  |  |  |  |
|-------------------------------------|----------------|-------|---|--|--|--|--|--|
| Halaman Data<br>Petugas<br>Operator | <i>Coding</i>  |       | 2 |  |  |  |  |  |
|                                     | <i>Testing</i> |       | 1 |  |  |  |  |  |
|                                     | Total          | 5 Jam |   |  |  |  |  |  |

**Tabel 3.19** *Sprint Backlog Sprint 13* Halaman Data User

| Sprint 13            | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|----------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                      |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Data<br>User | Membuat Database          | 1                    |   |   |   |   |   |   |
|                      | Tampilan <i>front-end</i> | 2                    |   |   |   |   |   |   |
|                      | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                      | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                      | Total                     | 6 Jam                |   |   |   |   |   |   |

**Tabel 3.20** *Sprint Backlog Sprint 14* Halaman Data Laporan Kebakaran

| Sprint 14                            | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|--------------------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                                      |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Data<br>Laporan<br>Kebakaran | Membuat Database          | 1                    |   |   |   |   |   |   |
|                                      | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                                      | <i>Coding</i>             |                      | 3 |   |   |   |   |   |
|                                      | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                                      | Total                     | 6 Jam                |   |   |   |   |   |   |

**Tabel 3.21** *Sprint Backlog Sprint 15* Halaman Login Operator

| Sprint 15                            | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|--------------------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                                      |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Login<br>Operator<br>Pemadam | Membuat Database          | 1                    |   |   |   |   |   |   |
|                                      | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                                      | <i>Coding</i>             |                      | 1 |   |   |   |   |   |
|                                      | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                                      | Total                     | 4 Jam                |   |   |   |   |   |   |

**Tabel 3.22** *Sprint Backlog Sprint 16* Halaman Profil Operator Pemadam

| Sprint 16                             | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|---------------------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                                       |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Profil<br>Operator<br>Pemadam | Membuat Database          | 1                    |   |   |   |   |   |   |
|                                       | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                                       | <i>Coding</i>             |                      | 1 |   |   |   |   |   |
|                                       | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                                       | Total                     | 4 Jam                |   |   |   |   |   |   |

**Tabel 3.23** *Sprint Backlog Sprint 17* Halaman Laporan Masuk

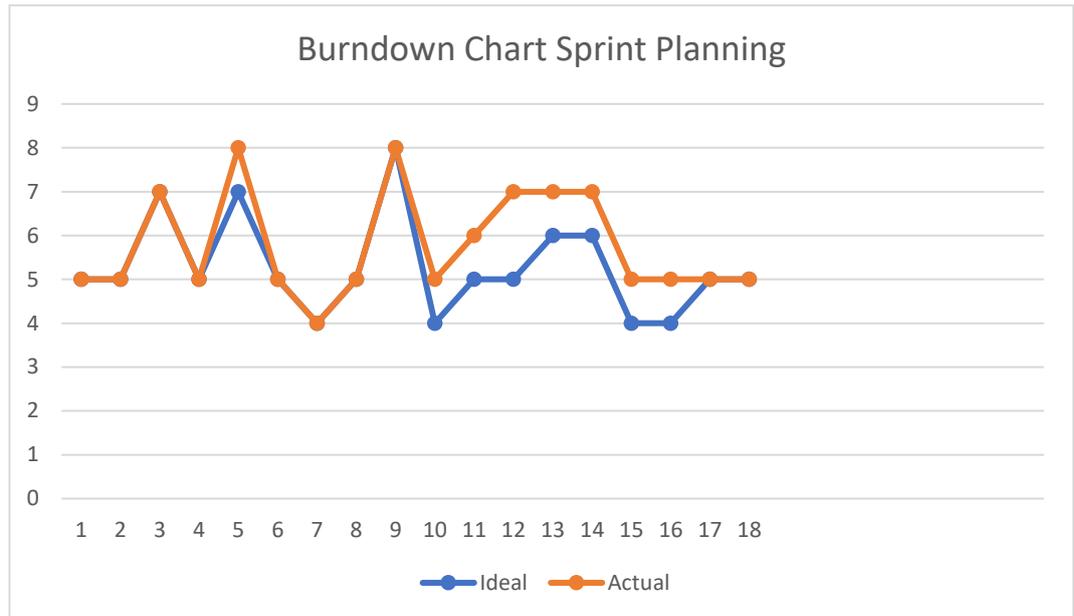
| Sprint 17             | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-----------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                       |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Laporan Masuk | Membuat Database          | 1                    |   |   |   |   |   |   |
|                       | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                       | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                       | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                       | Total                     | 5 Jam                |   |   |   |   |   |   |

**Tabel 3.24** *Sprint Backlog Sprint 18* Halaman Riwayat Laporan

| Sprint 18               | Task                      | Estimasi (Waktu/Jam) |   |   |   |   |   |   |
|-------------------------|---------------------------|----------------------|---|---|---|---|---|---|
|                         |                           | 1                    | 2 | 3 | 4 | 5 | 6 | 7 |
| Halaman Riwayat Laporan | Membuat Database          | 1                    |   |   |   |   |   |   |
|                         | Tampilan <i>front-end</i> | 1                    |   |   |   |   |   |   |
|                         | <i>Coding</i>             |                      | 2 |   |   |   |   |   |
|                         | <i>Testing</i>            |                      | 1 |   |   |   |   |   |
|                         | Total                     | 5 Jam                |   |   |   |   |   |   |

## 1.6 Daily Scrum

Berdasarkan *sprint planning* yang sudah dikerjakan, maka didapatkan grafik progress *burndown chart* yang ditampilkan saat *daily scrum*. *Burndown chart* merupakan laporan sprint dalam bentuk grafik garis dan digunakan sebagai pedoman dalam menentukan keberhasilan dari suatu sprint. Tujuan *daily scrum* adalah untuk memeriksa kemajuan menuju *sprint goal* dan mengadaptasi *Sprint Backlog* seperlunya dan menyesuaikan rencana pekerjaan yang akan datang.



**Gambar 3.46** *Burndown Chart Sprint Planning*

Gambar 3.46 menunjukkan bagaimana sprint planning dilaksanakan, diperkirakan 18 hari pengerjaan. Dari gambar 3.46, pengerjaan mengalami kemunduran terjadi pada hari ke-5, kemudian hari ke-10 hingga hari ke-16. Pada hari ke-17 grafik menunjukkan bahwa ideal dan actual sudah sesuai dan hari ke-18 tidak ada lagi estimasi pengerjaan, hal ini menunjukkan pekerja menyelesaikan sprint planning dengan baik.

### 1.7 Sprint Review

Setelah rangkaian *sprint* selesai dikerjakan, maka akan dilanjutkan ke tahap *sprint review* yang akan dilakukan demo untuk aplikasi yang sudah diimplementasikan. Aplikasi akan diperiksa secara berkala untuk mengetahui perkembangan yang terjadi dan mengevaluasi apakah produk yang dibangun sudah sesuai kebutuhan. Pengetesan aplikasi dilakukan menggunakan *black box testing* dengan *device smartphone* Xiaomi Redmi Note 11 dan Samsung A03S. Jika terjadi *error* atau *bug* maka aplikasi akan diperbaiki dan kembali melakukan *sprint* lagi dengan pemrograman. Namun jika aplikasi berjalan dengan lancar tanpa ada masalah, maka aplikasi sudah dinyatakan lulus uji dan siap untuk dipublikasikan. Pada *sprint review* ini akan dijelaskan di bab selanjutnya yaitu bab IV.