

RANCANG BANGUN APLIKASI HYBRID KONDISI JALAN DAN FASILITAS KELENGKAPAN JALAN DI KOTA MATARAM

Tugas Akhir
Untuk memenuhi sebagian persyaratan
Mencapai derajat Sarjana S-1 Program Studi Teknik Informatika



Oleh:
PUSPITA ASRI N.
F1D013087

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MATARAM
Juli 2020**

TUGAS AKHIR

RANCANG BANGUN APLIKASI HYBRID KONDISI JALAN DAN FASILITAS KELENGKAPAN JALAN DI KOTA MATARAM

Oleh:

**PUSPITA ASRI N.
F1D013087**

Telah diperiksa oleh Tim Pembimbing :

1. Pembimbing Utama



Nadivasari Agitha, S.Kom., M.MT.
NIP. 198608132018032001

Tanggal: 16/07/2020

2. Pembimbing Pendamping



Ahmad Zafrullah M., S.T., M.Eng.
NIP.

Tanggal: 08/07/2020

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Teknik
Universitas Mataram



Prof. Dr.Eng. I Gede Pasek Suta Wijaya, ST., MT.
NIP. 197311302000031001

TUGAS AKHIR

RANCANG BANGUN APLIKASI HYBRID KONDISI JALAN DAN FASILITAS KELENGKAPAN JALAN DI KOTA MATARAM

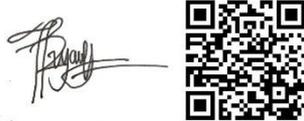
Oleh:

PUSPITA ASRI N.
F1D013087

Telah diujikan di depan penguji
Pada Tanggal 8 Juli 2020
Dan dinyatakan telah memenuhi syarat mencapai derajat Sarjana S-1
Program Studi Teknik Informatika

Susunan Tim Penguji :

1. Penguji 1



Tanggal: 18/07/2020

Royana Afwani, ST., MT.
NIP. 198507072014042001

2. Penguji 2



Tanggal: 15/07/2020

Moh. Ali Albar, ST., M.Eng.
NIP. 198311252015041002

3. Penguji 3



Tanggal: 17/07/2020

Ariyan Zubaidi, S.Kom., MT.
NIP. 198609132015041001

Mataram, 21 Juli 2020
Dekan Fakultas Teknik
Universitas Mataram



HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Saya yang bertanda tangan di bawah ini menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Mataram, 23 Juli 2020

Puspita Asri N.

PRAKATA

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat bimbingan, dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan Tugas Akhir dengan judul “*Rancang Bangun Aplikasi Hybrid Kondisi Jalan dan Fasilitas Kelengkapan Jalan di Kota Mataram*”.

Tugas Akhir ini dilaksanakan di Fakultas Teknik Universitas Mataram. Tujuan dari pembuatan Tugas Akhir ini adalah membuat aplikasi *hybrid* yang terdiri dari aplikasi *web* dan aplikasi *mobile* untuk memenuhi kebutuhan akan kecepatan dan keakuratan dalam melakukan pendataan, penyampaian ataupun untuk mendapatkan informasi terkait kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram. Tugas akhir ini juga merupakan salah satu persyaratan kelulusan guna mencapai gelar kesarjanaan di Program Studi Teknik Informatika, Fakultas Teknik Universitas Mataram.

Akhir kata semoga tidaklah terlampau berlebihan, bila penulis berharap agar karya ini dapat bermanfaat bagi pembaca.

Mataram, 23 Juli 2020

Penulis

UCAPAN TERIMAKASIH

Tugas Akhir ini dapat diselesaikan berkat bimbingan dan dukungan ilmiah maupun materil dari berbagai pihak, oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih yang setulus-tulusnya kepada :

1. Ibu Nadiyahari Agitha, S.Kom., M.MT. selaku dosen pembimbing utama yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir ini, sehingga dapat terselesaikan dengan baik.
2. Bapak Ahmad Zafrullah Mardiansyah, S.T., M.Eng. selaku dosen pembimbing pendamping yang telah memberikan bimbingan dan arahan selama penyusunan Tugas Akhir ini.
3. Ibu Royanan Afwani, ST., MT., Bapak Moh. Ali Albar, ST., M.Eng., dan Bapak Ariyan Zubaidi, S.Kom., MT. selaku dosen penguji yang telah memberikan kritik dan saran yang bersifat membangun dalam menyelesaikan Tugas Akhir ini.
4. Bapak Ida Bagus Ketut Widiartha, S.T., M.T yang juga telah memberikan arahan, bimbingan, dan banyak masukan yang sangat diperlukan dalam penyelesaian Tugas Akhir ini.
5. Keluarga penulis yang selalu memberikan dukungan selama proses mengerjakan Tugas Akhir ini.
6. Semua pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan bimbingan kepada penulis dalam menyelesaikan Tugas Akhir ini.

Semoga Tuhan Yang Maha Esa memberikan imbalan yang setimpal atas bantuan yang diberikan kepada penulis.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN TIM PEMBIMBING	ii
LEMBAR PENGESAHAN TIM PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
PRAKATA	v
UCAPAN TERIMAKASIH	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xv
INTISARI	xvi
ABSTRACT	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Sistematik Penulisan	4
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI	6
2.1 Tinjauan pustaka	6
2.2 Dasar Teori	7
2.2.1 Struktur Organisasi	8
2.2.2 Jalan	8
2.2.3 Perlengkapan Jalan	9
2.2.4 CodeIgniter	9
2.2.5 Ionic	9
2.2.6 <i>Unified Modeling Language (UML)</i>	10
2.2.7 <i>Entity Relationship Diagram (ERD)</i>	12
2.2.8 Metode <i>Black Box</i>	13
2.2.9 Kuesioner MOS	13

DAFTAR GAMBAR

Gambar 2.1 Diagram <i>fishbone</i>	7
Gambar 2.2 Struktur Organisasi Dinas Perhubungan Kota Mataram.	8
Gambar 2.3 Struktur Organisasi Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.	8
Gambar 2.4 Model pengembangan sistem waterfall.	13
Gambar 3.1 Model pembuatan aplikasi.	16
Gambar 3.2 Rancangan arsitektur aplikasi.	19
Gambar 3.3 <i>Use case</i> diagram aplikasi <i>hybrid</i>	20
Gambar 3.4 <i>Class</i> diagram sistem.	23
Gambar 3.5 <i>Sequence</i> diagram proses melihat pengaduan dan melakukam <i>feedback</i> . ..	24
Gambar 3.6 <i>Sequence</i> diagram proses mengelola pengumuman.	25
Gambar 3.7 <i>Sequence</i> diagram proses melihat kondisi jalan dan kelengkapan jalan.	26
Gambar 3.8 <i>Sequence</i> diagram proses melakukan pengaduan.	27
Gambar 3.9 <i>Activity</i> diagram proses <i>login admin</i>	28
Gambar 3.10 <i>Activity</i> diagram proses mengelola data.	29
Gambar 3.11 <i>Activity</i> diagram proses melihat laporan.	29
Gambar 3.12 <i>Activity</i> diagram proses melihat peta.	30
Gambar 3.13 <i>Activity</i> diagram proses melihat pengaduan.	30
Gambar 3.14 <i>Activity</i> diagram proses melihat kondisi.	31
Gambar 3.15 <i>Activity</i> diagram proses <i>login</i> dan <i>signup user</i>	32
Gambar 3.16 <i>Activity</i> diagram proses melakukan pengaduan.	32
Gambar 3.17 <i>Activity</i> diagram proses melihat <i>feedback</i>	33
Gambar 3.18 <i>Entity relationship</i> diagram aplikasi.	34
Gambar 3.19 Tampilan halaman menu peta untuk admin faskel.	39
Gambar 3.20 Tampilan halaman menu data papan nama.	39
Gambar 3.21 Tampilan halaman <i>form</i> data papan nama.	40
Gambar 3.22 Tampilan halaman menu data pemasangan papan nama.	40
Gambar 3.23 Tampilan halaman <i>form</i> pemasangan papan nama.	41
Gambar 3.24 Tampilan halaman menu data jenis rambu.	41
Gambar 3.25 Tampilan halaman <i>form</i> jenis rambu.	42

Gambar 3.26 Tampilan halaman menu data pemasangan rambu.	42
Gambar 3.27 Tampilan halaman <i>form</i> pemasangan rambu.	43
Gambar 3.28 Tampilan halaman menu data pemasangan APILL.	43
Gambar 3.29 Tampilan halaman <i>form</i> pemasangan APILL.	44
Gambar 3.30 Tampilan halaman menu data pemasangan RPPJ.	44
Gambar 3.31 Tampilan halaman <i>form</i> pemasangan RPPJ.	45
Gambar 3.32 Tampilan halaman menu pengaduan kerusakan faskel.	45
Gambar 3.33 Tampilan halaman detail pengaduan admin faskel.	46
Gambar 3.34 Tampilan halaman menu pengumuman admin faskel.	46
Gambar 3.35 Tampilan halaman <i>form</i> pengumuman admin faskel.	47
Gambar 3.36 Tampilan halaman menu laporan admin faskel.	47
Gambar 3.37 Tampilan halaman menu peta untuk admin jalan.	48
Gambar 3.38 Tampilan halaman menu data jalan.	48
Gambar 3.39 Tampilan halaman <i>form</i> data jalan.	49
Gambar 3.40 Tampilan halaman data kondisi jalan.	49
Gambar 3.41 Tampilan halaman <i>form</i> tambah data kondisi jalan.	50
Gambar 3.42 Tampilan halaman <i>form</i> edit data kondisi jalan.	50
Gambar 3.43 Tampilan halaman menu pengaduan kerusakan jalan.	51
Gambar 3.44 Tampilan halaman detail pengaduan admin jalan.	51
Gambar 3.45 Tampilan halaman menu pengumuman admin jalan.	52
Gambar 3.46 Tampilan halaman <i>form</i> pengumuman admin jalan.	52
Gambar 3.47 Tampilan halaman menu laporan admin jalan.	53
Gambar 3.48 Tampilan halaman <i>login</i> aplikasi <i>mobile</i>	53
Gambar 3.49 Tampilan halaman <i>sign up</i>	54
Gambar 3.50 Tampilan navigasi aplikasi.	54
Gambar 3.51 Tampilan halaman menu <i>home</i> dan <i>view</i> detail pengaduan.	55
Gambar 3.52 Tampilan halaman menu tab pengumuman.	55
Gambar 3.53 Tampilan tab menu faskel jalan.	56
Gambar 3.54 Tampilan halaman menu pengaduan.	56
Gambar 3.55 Tampilan form pengaduan kerusakan.	57
Gambar 3.56 Tampilan halaman menu pengaduanku.	57
Gambar 3.57 Tampilan halaman menu pengaturan akun.	58
Gambar 4.1 Daftar tabel pada <i>database</i> sistem.	61

Gambar 4.2 Daftar <i>class</i> model pada sistem.....	62
Gambar 4.3 Daftar <i>class controller</i> pada <i>project website</i>	63
Gambar 4.4 Daftar <i>class view</i> pada <i>project website</i>	63
Gambar 4.5 Daftar <i>class controller</i> pada <i>project mobile</i>	64
Gambar 4.6 Daftar <i>class view</i> pada <i>project mobile</i>	64
Gambar 4.7 <i>Site map</i> sistem.	65
Gambar 4.8 <i>Interface</i> menu peta untuk <i>admin</i> jalan.....	66
Gambar 4.9 <i>Interface</i> menu data jalan.....	67
Gambar 4.10 <i>Interface</i> menu data jalan.....	67
Gambar 4.11 <i>Interface</i> menu data kondisi jalan.	68
Gambar 4.12 <i>Interface</i> detail pengaduan kerusakan jalan.....	68
Gambar 4.13 <i>Interface</i> edit data pengumuman jalan.	69
Gambar 4.14 <i>Interface</i> menu laporan untuk <i>admin</i> jalan.	70
Gambar 4.15 <i>Interface</i> menu peta untuk <i>admin</i> fasilitas kelengkapan jalan.	71
Gambar 4.16 <i>Interface</i> menu data pemasangan papan nama jalan.....	71
Gambar 4.17 <i>Interface</i> menu data pemasangan rambu lalu lintas.....	72
Gambar 4.18 <i>Interface</i> menu data pemasangan APILL.	72
Gambar 4.19 <i>Interface</i> menu data pemasangan RPPJ.	73
Gambar 4.20 <i>Interface</i> menu pengaduan kerusakan fasilitas kelengkapan jalan.	74
Gambar 4.21 <i>Interface</i> menu pengumuman fasilitas kelengkapan jalan.....	74
Gambar 4.22 <i>Interface</i> menu laporan untuk <i>admin</i> fasilitas kelengkapan jalan.....	75
Gambar 4.23 <i>Interface</i> menu <i>home</i> untuk <i>tab list</i> pengaduan (kiri) dan detail penagduan(kanan).....	75
Gambar 4.24 <i>Interface</i> menu <i>home</i> untuk <i>tab list</i> pengumuman (kiri) dan detail pengumuman(kanan).	76
Gambar 4.25 <i>Interface</i> menu <i>home</i> untuk <i>tab list</i> faskel (kiri) dan detail faskel(kanan).	76
Gambar 4.26 <i>Interface</i> menu <i>home</i> -peta.	77
Gambar 4.27 kategori pengaduan kerusakan.....	77
Gambar 4.28 <i>Interface</i> peta (kiri) dan <i>form</i> penagduan kerusakan jalan (kanan).	78
Gambar 4.29 <i>Interface</i> peta (kiri) dan <i>form</i> penagduan kerusakan fasilitas kelengkapan jalan (kanan).	78
Gambar 4.30 <i>Interface</i> menu pengaduanku.....	79

Gambar 4.31 <i>Interface</i> menu pengaturan akun.....	79
Gambar 4.32 Grafik rata-rata responden Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.	86
Gambar 4.33 Grafik rata-rata responden Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.	87
Gambar 4.34 Grafik rata-rata responden masyarakat.	88

DAFTAR TABEL

Tabel 2.1 Simbol <i>class</i> diagram.....	10
Tabel 2.2 Simbol <i>use case</i> diagram.	11
Tabel 2.3 Simbol <i>activity</i> diagram.....	11
Tabel 2.4 Simbol <i>sequence</i> diagram.	12
Tabel 2.5 Simbol <i>entity relationship</i> diagram.	12
Tabel 2.6 Standar penilaian MOS.....	13
Tabel 3.1 Struktur tabel admin.	35
Tabel 3.2 Struktur tabel mobile_user.....	35
Tabel 3.3 Struktur tabel jalan.....	35
Tabel 3.4 Struktur tabel kondisi_jalan.....	35
Tabel 3.5 Struktur tabel jenis_rambu.....	36
Tabel 3.6 Struktur tabel pasang_rambu.	36
Tabel 3.7 Struktur tabel papan_nama.	36
Tabel 3.8 Struktur tabel pasang_pn.	36
Tabel 3.9 Struktur tabel pasang_apill.	37
Tabel 3.10 Struktur tabel pasang_rppj.....	37
Tabel 3.11 Struktur tabel pengumuman	37
Tabel 3.12 Struktur tabel pengaduan.	37
Tabel 3.13 Struktur tabel <i>feedback</i>	38
Tabel 3.14 Struktur tabel melakukan_pengaduan.....	38
Tabel 3.15 Rencana pengujian <i>black box</i>	59
Tabel 4.1 Pengujian fungsi tambah pengumuman.....	80
Tabel 4.2 Pengujian fungsi edit pengumuman.....	81
Tabel 4.3 Pengujian fungsi hapus pengumuman.	81
Tabel 4.4 Pengujian fungsi hapus pengumuman.	82
Tabel 4.5 Pengujian fungsi melihat fasilitas kelengkapan jalan.....	82
Tabel 4.6 Pengujian fungsi melihat pengumuman.....	83
Tabel 4.7 Pengujian fungsi melihat pengaduan.....	83
Tabel 4.8 Pengujian melakukan pengaduan kerusakan.	84

Tabel 4.9 hasil pengujian MOS Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.	85
Tabel 4.10 hasil pengujian MOS Dinas Perhubungan Kota Mataram.	86
Tabel 4.11 hasil pengujian MOS masyarakat Kota Mataram.	87

DAFTAR LAMPIRAN

- Lampiran 1 *Sequence* Diagram
- Lampiran 2 Implementasi Tabel pada *Database*
- Lampiran 3 Hasil Pengujian Metode *Black Box*

INTISARI

Aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di kota Mataram merupakan aplikasi yang terdiri dari aplikasi *web* dan *mobile* yang saling berintegrasi. Aplikasi *web* digunakan oleh Dinas Pekerjaan Umum dan Penataan Ruang kota Mataram dan Dinas Perhubungan Kota Mataram untuk mengelola data kondisi dari ruas jalan kota dan juga kondisi fasilitas kelengkapan jalan yang ada di kota Mataram dan untuk menerima pengaduan kerusakan oleh masyarakat. Sedangkan aplikasi *mobile* digunakan oleh masyarakat kota Mataram untuk melihat kondisi dari jalan dan fasilitas kelengkapan jalan berdasarkan data dari dinas dan masyarakat juga dapat melakukan pengaduan kerusakan jalan dan fasilitas kelengkapan jalan. Pengujian sistem menggunakan metode Black Box dan pengujian MOS. Hasil uji MOS menunjukkan bahwa rata-rata penilaian responden dari Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram menyatakan sangat setuju dan setuju adalah 66,67% dan 33,33%. Rata-rata penilaian responden dari Dinas Perhubungan Kota Mataram menyatakan sangat setuju dan setuju adalah 58,33% dan 41,67%. Sedangkan, rata-rata penilaian responden dari masyarakat menyatakan sangat setuju dan setuju adalah 52,50% dan 44,17%. Oleh karena itu, dapat disimpulkan bahwa sistem ini layak untuk digunakan.

Kata kunci : Aplikasi *Hybrid*, Jalan, Fasilitas Kelengkapan Jalan, Pengaduan, MOS.

ABSTRACT

Hybrid Application of Roads and Road Fittings Facilities Conditions in the City of Mataram is an application consisting of web applications and mobile applications. The web application is used by Dinas Pekerjaan Umum dan Penataan Ruang of mataram city and Dinas Perhubungan of mataram city to manage condition data from city roads and also road fittings facilities in the city of Mataram and to receive complaints of damage from the public. While the mobile application is used by the people of the city of Mataram to see the condition of the road and road fittings facilities based on data from the department and the public can also make complaints about damage from roads and road facilities. System testing using the Black Box method and MOS testing. MOS test results indicate that the average rating of respondents from Dinas Pekerjaan Umum dan Penataan Ruang of Mataram City stated strongly agree and agree were 66.67% and 33.33%. The average rating of respondents from Dinas Perhubungan of Mataram City stated strongly agree and agree were 58,33% and 41,67%. The average rating of respondents from the community stated strongly agree and agree were 52,50% and 44,17%. Therefore, this research concluded that this system is eligible to use.

Keywords : *Hybrid Application, Road, Road Fittings Facilities, complaint, MOS.*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Jalan merupakan prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap dan perlengkapannya yang diperuntukan bagi lalu lintas. Sedangkan, perlengkapan jalan adalah sarana yang dimaksudkan untuk keselamatan, keamanan, ketertiban, dan kelancaran lalu lintas serta kemudahan bagi pengguna jalan dalam lalu lintas [1]. Jalan yang terpasang perlengkapan jalan bertujuan untuk mencegah tindakan-tindakan yang membahayakan keselamatan lalu lintas, sehingga sangat penting bagi lembaga pemerintah terkait bekerja sama dengan masyarakat pengguna jalan maupun penduduk sekitar jalan untuk memperhatikan kondisi jalan dan perlengkapannya.

Jalan dan kelengkapan jalan di Kota Mataram ditangani oleh beberapa lembaga pemerintah yang berbeda. Untuk memantau kondisi ruas jalan dengan status jalan kota dilakukan oleh Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram sedangkan untuk beberapa fasilitas kelengkapan jalan di kota Mataram ini ditangani oleh Dinas Perhubungan Kota Mataram. Dari kedua lembaga tersebut dalam pengelolaan datanya masih belum memanfaatkan sebuah sistem yang saling berintegrasi, dimana data-data dalam setahun tersimpan dalam sebuah *file* berformat *document*. Hal tersebut dapat mengakibatkan kemungkinan kehilangan data yang sangat besar dan juga kesulitan dalam menganalisa kesimpulan data untuk membuat laporan tahunan. Begitu juga dengan informasi kegiatan perbaikan/pembangunan jalan ataupun fasilitas kelengkapan jalan yang sedang dilakukan sering ditemukan oleh masyarakat hanya pada akun media sosial milik dinas atau melalui media berita. Maka diperlukan suatu sistem yang terintegrasi untuk mengelola data yang banyak dari setiap ruas jalan dan fasilitas kelengkapan jalan yang terpasang, serta mengelola pengumuman perbaikan/pembangunan yang dilakukan dinas sehingga menjadi informasi yang dapat diakses oleh masyarakat dan juga masyarakat dapat melakukan pengaduan kerusakan yang terkirim langsung ke dinas yang menangani jalan maupun menangani fasilitas kelengkapan jalan.

Seiring dengan berkembangnya teknologi informasi, banyak teknologi baru yang muncul seperti halnya teknologi *mobile* berbasis *android* yang memudahkan pengguna dalam mengakses informasi yang dibutuhkan dimana saja dan kapan saja. Selain itu, terdapat *hybrid application* sebagai salah satu jenis aplikasi *mobile* yang merupakan

aplikasi web yang ditransformasikan menjadi kode *native* pada *platform mobile*. Jenis aplikasi *mobile* tersebut dipasang langsung pada perangkat *mobile* namun menjalankannya membutuhkan koneksi internet, serta memiliki kelebihan dimana aplikasi dapat dijalankan pada *platform* yang berbeda-beda. Selain itu, terdapat perkembangan suatu Sistem Informasi Geografis (SIG) yang dapat didefinisikan sebagai sistem komputer yang digunakan untuk memanipulasi, menganalisis, dan menampilkan data-data yang berhubungan dengan posisi-posisi di permukaan bumi.

Berdasarkan uraian di atas, maka dalam penelitian ini akan dirancang dan dibangun aplikasi *hybrid* untuk memenuhi kebutuhan akan kecepatan dan keakuratan dalam melakukan pendataan, penyampaian ataupun untuk mendapatkan informasi terkait kondisi jalan dan fasilitas kelengkapan jalan yang berada di Kota Mataram. Pada aplikasi yang akan digunakan oleh *admin*, berupa aplikasi *web* dengan fitur mengelola data jalan dan fasilitas kelengkapan, pengumuman perencanaan pemasangan atau perbaikan yang akan dilakukan, melihat pengaduan-pengaduan kerusakan, melakukan *feedback* dari pengaduan yang diterima, dan melihat laporan tahunan berupa tabel dan grafik untuk pengaduan kerusakan, kondisi jalan, dan pengadaan fasilitas kelengkapan jalan. Aplikasi yang akan digunakan oleh *user* (masyarakat Kota Mataram), berupa aplikasi *mobile* dengan fitur melihat kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram, melihat pengumuman pembangunan atau perbaikan akan dilakukan, dan melihat pengaduan yang telah dilakukan oleh *user* lain. Fitur aplikasi lainnya yaitu untuk mengirim pengaduan berdasarkan kategori kerusakan jalan ataupun kerusakan kelengkapan jalan, disertai dengan mengirimkan lokasi keberadaan fasilitas yang rusak dan bukti berupa gambar.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang yang ada, maka rumusan masalah dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana merancang dan membuat aplikasi *web* dari sisi dinas yang bertanggung jawab masalah jalan ataupun kelengkapan jalan untuk mempermudah mengelola data, memberikan informasi kepada masyarakat mengenai kondisi jalan dan kondisi fasilitas kelengkapan jalan yang ada di Kota Mataram, serta untuk menerima pengaduan kerusakan dari masyarakat ?

2. Bagaimana merancang dan membuat aplikasi *mobile* yang mempermudah masyarakat Kota Mataram melihat kondisi jalan dan fasilitas kelengkapan jalan, serta mengirim pengaduan kerusakan kepada dinas yang bertanggung jawab ?

1.3 Batasan Masalah

Berdasarkan rumusan masalah yang ada, maka diuraikan beberapa batasan masalah dalam tugas akhir ini adalah sebagai berikut:

1. Sistem yang akan dibangun terdiri dari aplikasi *mobile* yang akan digunakan oleh masyarakat Kota Mataram sebagai *mobile user* dan aplikasi *web* digunakan oleh Dinas Perhubungan Kota Mataram serta Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram sebagai *admin*.
2. Aplikasi *web* akan digunakan oleh *admin* untuk mengelola data, serta memberikan informasi penting terkait program pembangunan atau perbaikan yang akan dilakukan, dan untuk menerima pengaduan kerusakan dari masyarakat.
3. Aplikasi *mobile* akan digunakan oleh *mobile user* untuk mengetahui kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram, serta dapat melakukan pengaduan berkaitan dengan adanya jalan rusak atau fasilitas kelengkapan jalan yang rusak.
4. Berdasarkan data yang diperoleh Dinas Perhubungan Kota Mataram, yang termasuk kelengkapan jalan yang akan ditampilkan dalam aplikasi ini antara lain yaitu rambu-rambu jalan, papan nama jalan, Rambu Pendahulu Petunjuk Jurusan (RPPJ), dan Alat Pemberi Isyarat Lalu Lintas (APILL).
5. Berdasarkan data yang diperoleh dari Bina Marga pada Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram, informasi jalan yang akan dikelola berupa karakteristik ruas jalan kota yaitu nama ruas jalan, panjang ruas, jenis permukaan, lebar ruas, kecamatan yang dilalui, status jalan, dan kondisi terakhir jalan.
6. Dibutuhkan koneksi internet dengan sinyal yang sangat kuat untuk dapat melakukan pengiriman pengaduan dengan posisi kerusakan yang akurat serta untuk menerima pengaduan kerusakan.
7. Fitur aplikasi yang akan dibuat tidak mencakup fitur keamanan dari aplikasi seperti halnya untuk mencegah terdapatnya akun palsu.

1.4 Tujuan Penelitian

Adapun tujuan dari pembuatan tugas akhir ini adalah sebagai berikut:

1. Merancang dan membuat aplikasi *web* dari sisi *admin* untuk mempermudah mengelola data, memberikan informasi kepada masyarakat mengenai kondisi jalan dan kondisi kelengkapan jalan, serta untuk menerima pengaduan kerusakan dari masyarakat.
2. Merancang dan membangun aplikasi *mobile* yang mempermudah *mobile user* melihat kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram dalam bentuk peta Kota Mataram beserta marker yang menandakan posisi fasilitas kelengkapan yang terpasang, posisi perbaikan yang akan dilakukan, dan posisi pengaduan kerusakan yang pernah dilakukan. Selain itu, *user* dapat mengirim pengaduan kerusakan kepada *admin*.

1.5 Manfaat Penelitian

Adapun manfaat dari pembuatan tugas akhir ini adalah sebagai berikut:

1. Dapat memudahkan dinas yang bertanggung jawab masalah jalan maupun fasilitas kelengkapan jalan dalam mengelola data dan memberikan informasi kegiatan perbaikan yang akan dilakukan serta memperoleh informasi lokasi jalan rusak ataupun lokasi kelengkapan jalan yang rusak yang berada di Kota Mataram.
2. Dapat memudahkan masyarakat Kota Mataram untuk melihat kondisi jalan dan fasilitas kelengkapan jalan yang terpasang di Kota Mataram, serta untuk melakukan pengaduan kerusakan jalan dan fasilitas kelengkapan jalan di Kota Mataram dimana saja.

1.6 Sistematik Penulisan

Untuk memudahkan dan teraturnya penyusunan Tugas Akhir ini, maka penulis menyusun secara garis besar dalam 5 (lima) bab yaitu sebagai berikut:

BAB I PENDAHULUAN

Bab ini mendeskripsikan masalah yang melatarbelakangi perancangan dan pembangunan aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram, mendefinisikan batasan-batasan masalah yang akan dikerjakan dalam tugas akhir ini, tujuan dari pembuatan aplikasi serta manfaat yang didapatkan dari aplikasi yang dibuat pada tugas akhir.

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bab ini terdiri dari tinjauan pustakan peneliti terdahulu yang digunakan sebagai acuan sejauh mana aplikasi yang menangani masalah pengelolaan data jalan dan fasilitas kelengkapan jalan telah dikembangkan. Selain itu juga terdapat dasar teori tentang jalan dan perlengkapan jalan, ionic, UML, CodeIgniter, *Entity Relationship Diagram*, pengujian dengan metode *black box testing* dan kuesioner MOS sebagai acuan dalam pembuatan tugas akhir.

BAB III METODE PENELITIAN

Bab ini dibuat alur metode penelitian yang akan dilakukan sampai selesai. Metode penelitian terdiri dari pengumpulan data, analisis, perancangan aplikasi, *coding* aplikasi, pengujian aplikasi dan dokumentasi.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memuat tentang hasil dari aplikasi yang di rancang beserta fungsi dari masing-masing fitur dan data hasil pengujian dari aplikasi yang telah di lakukan.

BAB V PENUTUP

Bab ini memuat tentang kesimpulan dari tugas akhir dan saran yang untuk pengembangan selanjutnya.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

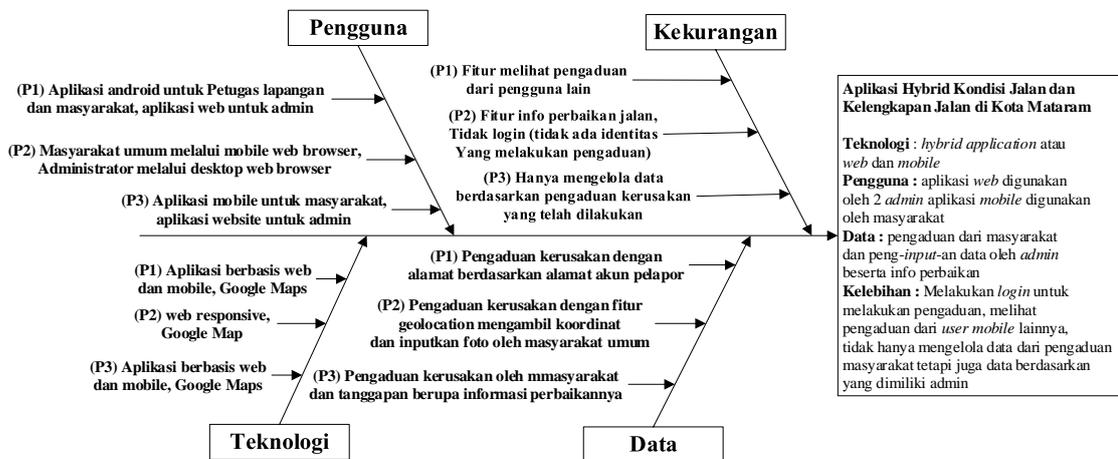
2.1 Tinjauan pustaka

Nastiar [2] merancang dan membangun sistem pelaporan kerusakan jalan berbasis *web* dan *mobile*, laporan dalam bentuk foto langsung dapat dikirim ke *website* dan petugas dapat menampilkan langsung laporan yang telah diterima kepada kepala bidang Bina Marga. Sistem memudahkan dalam pengolahan data pelaporan kerusakan jalan, pemetaan laporan menggunakan *maps* tersimpan dengan baik yang memudahkan dalam proses pencarian data. Adapun kekurangan dalam penelitian ini adalah pengguna aplikasi hanya dapat melihat pelaporan kerusakan jalan yang pernah dilakukannya tapi tidak dapat melihat kerusakan jalan yang dilaporkan oleh pengguna lainnya.

Sihombing [3] merancang sistem informasi pelaporan masyarakat dengan dua antarmuka yang *responsive*, dimana dapat dibuka melalui dekstop *web browser* maupun *mobile web browser*. Sistem dapat menampilkan informasi pemetaan kerusakan jalan yang ada di Pontianak menggunakan Google Maps API, menampilkan secara *real time* koordinat pengguna dari perangkat GPS yang ada pada *smartphone* pengguna saat melakukan pelaporan kerusakan, dan administrator dapat menerima laporan kerusakan dari masyarakat berupa foto dan data kerusakan jalan. Adapun kekurangan dalam penelitian ini adalah sistem hanya dimanfaatkan untuk melakukan pengaduan kerusakan jalan saja, tidak ada informasi yang diberikan tentang perbaikan jalan yang akan dilakukan. Selain itu, karena masyarakat melakukan pengaduan kerusakan tanpa melalui *login* dahulu maka menyulitkan untuk mencari pengaduan yang pernah dilakukan.

Pramuji [4] merancang aplikasi android menggunakan bahasa pemrograman java, aplikasi ini dapat melakukan pengaduan laporan kerusakan jalan yang berada di wilayah Sleman secara *real*, menampilkan informasi dari dinas Pekerja Umum dan Perumahan tentang jalan seperti informasi perbaikan jalan yang dilakukan atau sedang dilakukan oleh dinas Pekerja Umum dan Perumahan. Aplikasi dapat menghasilkan laporan berupa grafik tingkat kerusakan jalan per kecamatan di wilayah Sleman. Adapun kekurangan dalam penelitian ini, dimana aplikasi *web* yang digunakan oleh *admin* yang hanya mengelola data kerusakan jalan yang berasal dari laporan yang telah dilakukan masyarakat, begitupun informasi umum yang diberikan ke masyarakat merupakan informasi yang merupakan tanggapan *admin* dari laporan kerusakan yang telah dilakukan masyarakat.

Tinjauan pustaka dari penelitian-penelitian tersebut digambarkan dalam sebuah diagram *fishbone* pada Gambar 2.1.



Gambar 2.1 Diagram *fishbone*.

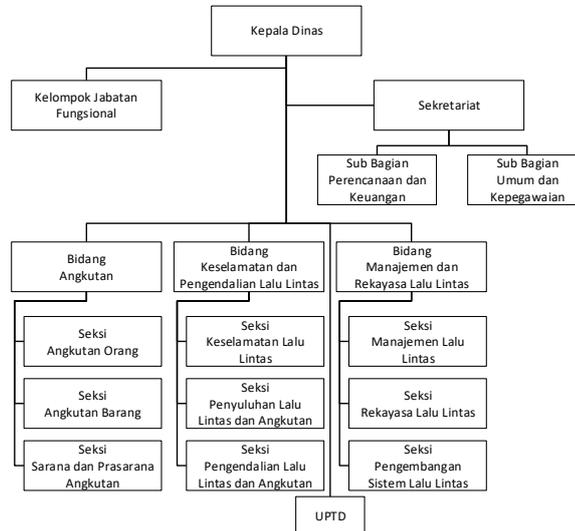
Berdasarkan tinjauan di atas, maka pembahasan pada tugas akhir ini adalah bagaimana merancang serta membangun aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram. Perancangan memiliki beberapa kesamaan dan kelebihan dari penelitian sebelumnya. Sistem yang dirancang berbasis *mobile* dan *web* yang memanfaatkan teknologi Google Maps. Pengguna sistem terdiri dari masyarakat kota Mataram sebagai pengguna sistem berbasis *mobile*, serta dinas yang menangani jalan dan dinas yang menangani fasilitas kelengkapan jalan sebagai pengguna sistem berbasis *web*. Sistem berbasis *web* memiliki fitur mengelola data jalan dan mengelola data fasilitas kelengkapan jalan yang ada di Kota Mataram, memberikan informasi perbaikan atau pembangunan yang memungkinkan dapat mengganggu kenyamanan pengguna jalan dan masyarakat sekitar, fitur melihat pengaduan kerusakan yang dilakukan masyarakat kota, dan melihat laporan tahunan yang berdasarkan pengaduan kerusakan yang dilakukan masyarakat dan juga kondisi dari jalan dan fasilitas kelengkapan jalan sesuai dari pendataan yang telah dilakukan. Untuk sistem berbasis *mobile* memiliki fitur untuk melakukan pengaduan kerusakan disertai dengan bukti berupa foto dan koordinat kerusakan, fitur melihat pengaduan yang dilakukan oleh pengguna aplikasi *mobile* lain, fitur melihat pengumuman perbaikan atau pembangunan yang dilakukan, fitur melihat fasilitas kelengkapan jalan yang terpasang.

2.2 Dasar Teori

Pada bagian ini, akan dibahas tentang dasar teori berkaitan dengan konsep-konsep dan materi yang mendukung perancangan aplikasi.

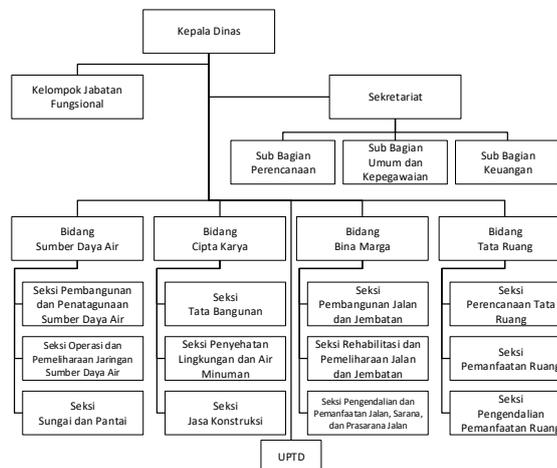
2.2.1 Struktur Organisasi

Dalam menyelesaikan tugas akhir ini, proses penelitian dan pengambilan data dilakukan di Dinas Perhubungan Kota Mataram dan Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.



Gambar 2.2 Struktur Organisasi Dinas Perhubungan Kota Mataram.

Pada bidang Manajemen dan Rekayasa Lalu Lintas pada Dinas Perhubungan Kota Mataram yang memiliki salahsatu tugas dalam pelaksanaan pengawasan fasilitas kelengkapan sarana prasaranan LLAJ (Lalu Lintas dan Angkutan Jalan).



Gambar 2.3 Struktur Organisasi Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.

Pada bidang Bina Marga pada Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram yang memiliki salahsatu tugas dalam pemeliharaan jalan dan jembatan.

2.2.2 Jalan

Jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap dan perlengkapan yang diperuntukan bagi lalu lintas, yang

berada pada permukaan tanah, di atas permukaan tanah, di bawah permukaan tanah atau air, serta di atas permukaan air, kecuali jalan kereta api, jalan lori, dan jalan kabel [1].

2.2.3 Perlengkapan Jalan

Perlengkapan jalan adalah sarana yang dimaksudkan untuk keselamatan, keamanan, ketertiban, dan kelancaran lalu lintas serta kemudahan bagi pengguna jalan dalam berlalu lintas yang meliputi marka jalan, rambu lalu lintas, alat pemberi isyarat lalu lintas, lampu penerang jalan, rel pengaman, dan penghalang lalu lintas [1].

2.2.4 CodeIgniter

CodeIgniter memudahkan *developer* untuk membuat aplikasi *web* dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. CodeIgniter merupakan aplikasi *open source* yang berupa *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. MVC adalah sebuah pendekatan yang ditempuh untuk memisahkan aplikasi menjadi 3 bagian, yaitu *Model, View* dan *Controller*. MVC memberikan struktur kepada aplikasi, sehingga dapat dicapai “*code reusability*”.

1. Model

Model merepresentasikan data yang digunakan aplikasi. Contoh data misalnya *database*, RSS, atau data yang diperoleh dari pemanggilan API, dan aksi yang melibatkan operasi *Create, Read, Update* dan *Delete* (CRUD) data.

2. View

View adalah informasi yang ditampilkan kepada *user* melalui *browser*. Informasi yang ditampilkan biasanya berupa *file* HTML atau kode PHP yang menyusun *template* untuk sebuah *website*.

3. Controller

Controller bertugas sebagai jembatan antara model dan *view*. *Controller* akan merespon HTTP *request* yang datang dari *user* melalui *browser*, dari *request* ini *controller* akan menentukan proses yang akan dilakukan [5].

2.2.5 Ionic

Ionic framework adalah kerangka kerja yang dibangun untuk mengembangkan aplikasi *hybrid mobile* dengan menggunakan teknologi *web*. Ionic menyediakan komponen-komponen *mobile* seperti *button, header-footer, cards*, dan lain sebagainya

kemudian menyediakan css standar dan menggunakan javascript untuk menjalankan fungsi-fungsi aplikasi, untuk javascript sendiri ionic memakai teknologi angular.js [6].

2.2.6 Unified Modeling Language (UML)

UML (*Unified Modeling Language*) adalah bahasa yang menggambarkan cara kerja maupun struktur dari suatu sistem yang dibuat terlebih menggunakan konsep OOP (*Object Oriented Programming*). Dengan UML, sistem dapat dilihat dari struktur data yang diberikan, tingkah laku apa saja dari suatu objek hingga aktivitas-aktivitas yang ada di dalamnya. UML sendiri sangat membantu sekali bagi para pengembang dalam membuat rancangan sebelum program tersebut dibuat dan membantu dalam pembuatan dokumentasi [7]. Pada UML itu sendiri terdapat beberapa diagram yaitu *class* diagram, *use case* diagram, *sequence* diagram dan *activity* diagram. Uraian untuk diagram-diagram tersebut adalah sebagai berikut:

1. *Class* diagram

Class diagram menggambarkan bagaimana objek pada dunia nyata digambarkan pada struktur yang biasa memiliki atribut dan *method*. Pada umumnya *class* diagram juga dapat menunjukkan hubungan antara relasi dari setiap *class* yang dibuat dan memperlihatkan objek – objek yang menyusun dari sistem tersebut. Pada *class* diagram terdapat beberapa simbol diantaranya dapat dilihat pada Tabel 2.1.

Tabel 2.1 Simbol *class* diagram.

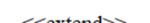
Simbol	Nama	Keterangan
	<i>Class</i>	<i>Class</i> adalah sebuah objek yang menggambarkan sebuah keadaan nyata. <i>Class</i> memiliki : Nama kelas, atribut, dan <i>method</i> .
	<i>Association</i>	Menunjukkan relasi antar <i>class</i> dengan makna umum.
	<i>Direction Association</i>	Menunjukkan relasi antar <i>class</i> dengan makna <i>class</i> yang satu digunakan oleh <i>class</i> yang lain.
	<i>Generalization</i>	Menunjukkan relasi antar <i>class</i> dengan makna generalisasi-spesialisasi (umum khusus). Disebut juga <i>inheritance</i> (pewarisan).

2. *Use Case* diagram

Use case mendeskripsikan interaksi antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan. *Use case* diagram menampilkan aktor mana yang menggunakan *use case* mana, *use case* mana yang memerlukan *use case* lain dan hubungan antara aktor dan *use case*.

Pada *use case* diagram terdapat beberapa simbol yang digunakan diantaranya dapat dilihat pada Tabel 2.2.

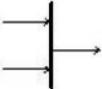
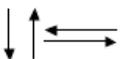
Tabel 2.2 Simbol *use case* diagram.

Simbol	Nama	Keterangan
	<i>Actor</i>	<i>Actor</i> tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
	<i>Use Case</i>	Gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.
	<i>Association</i>	Sebagai penghubung antara <i>actor</i> dengan <i>use case</i> yang saling berinteraksi.
	<i>Include</i>	Menggambarkan hubungan dimana <i>use case</i> tidak dapat berdiri sendiri, artinya selalu memerlukan <i>use case</i> yang lainnya.
	<i>extend</i>	Menggambarkan hubungan dimana <i>use case</i> target merupakan perluasan dari <i>use case</i> pangkal
	<i>Generalization</i>	Disebut juga <i>inheritance</i> (pewarisan).

3. *Activity* diagram

Activity diagram adalah cara untuk menggambarkan aktivitas – aktivitas yang terdapat pada suatu sistem yang akan dibuat. Dengan *activity* diagram, alur dari kerja suatu sistem dapat dibuat dari awal hingga akhir. Pada *Activity* diagram terdapat beberapa simbol diantaranya dapat dilihat pada Tabel 2.3.

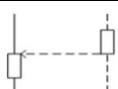
Tabel 2.3 Simbol *activity* diagram.

Simbol	Nama	Keterangan
	<i>Initial Activity</i>	Menggambarkan titik awal aktivitas sistem.
	<i>Activity</i>	Menunjukkan aktivitas yang dilakukan
	<i>Final Activity</i>	Menggambarkan titik akhir aktivitas sistem.
	<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan/tindakan yang harus diambil pada kondisi tertentu.
	<i>Join</i>	Menunjukkan penggabungan kegiatan paralel menjadi satu.
	<i>Fork</i>	Menunjukkan percabangan kegiatan yang dilakukan secara paralel.
	<i>Controll Flow</i>	Memperlihatkan urutan – urutan eksekusi sistem.

4. Sequence diagram

Sequence diagram adalah grafik dua dimensi dimana obyek ditunjukkan dalam dimensi horizontal. Fungsi dari *sequence* diagram adalah untuk menunjukkan rangkaian pesan yang dikirimkan antara objek dalam kelas dan interaksi antar objek yang dideskripsikan dalam urutan dari eksekusi. *Sequence* diagram terdapat beberapa simbol diantaranya dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol *sequence* diagram.

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
	<i>Object</i>	Sebuah objek yang berpartisipasi secara berurutan dengan mengirimkan dan/atau menerima pesan.
	<i>Lifeline</i>	Eksekusi objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya).
	<i>Object active</i>	Menandakan ketika suatu objek mengirim atau menerima pesan.
	<i>Message</i>	Objek mengirim satu pesan ke objek lainnya.
	<i>Return Value</i>	Menggambarkan hasil dari pengiriman pesan. Objek menghasilkan suatu kembalian ke objek tertentu.

2.2.7 Entity Relationship Diagram (ERD)

ERD (*Entity Relationship Diagram*) merupakan suatu model yang digunakan untuk menjelaskan hubungan antar data dalam basis data dan relasi yang dimiliki, selain itu ERD juga memodelkan struktur data dimana untuk menggambarkannya digunakan simbol dan notasi [8]. Beberapa simbol yang digunakan dalam ERD dapat dilihat pada Tabel 2.5.

Tabel 2.5 Simbol *entity relationship diagram*.

Simbol	Nama	Keterangan
	<i>Entity</i>	<i>Entity</i> adalah objek dalam dunia nyata yang akan digambarkan dalam lingkungan pemakai.
	<i>Relationship</i>	<i>Relationship</i> menunjukkan adanya hubungan antara entitas yang berbeda.
	<i>Attribute</i>	<i>Attribute</i> memberikan deskripsi dari entitas.
	<i>Key Attribute</i>	<i>Attribute</i> unik yang mewakili dari sebuah entitas.

2.2.8 Metode *Black Box*

Black box terfokus pada apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. Pada *black box testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul. Kemudian diamati apakah hasil dari unit sesuai dengan proses bisnis yang diinginkan [8].

2.2.9 Kuesioner MOS

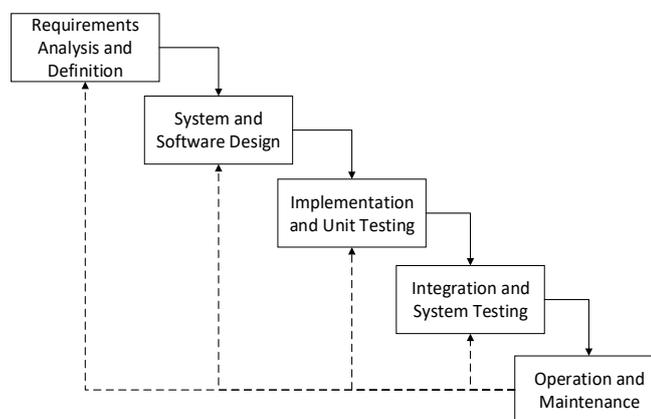
Kuesioner adalah suatu teknik pengumpulan informasi yang memungkinkan analisis untuk mempelajari karakteristik dari sistem yang telah ada. Kuisisioner melibatkan responden di dalam sistem untuk memberikan penilaian subjektif dengan metode MOS. Definisi *Mean Opinion Score* adalah nilai skalar yang telah ditetapkan pada subjek yang diteliti sebagai opini performa dari sistem transmisi [9]. Standar penilaian MOS dapat dilihat pada Tabel 2.6.

Tabel 2.6 Standar penilaian MOS.

Nilai MOS	Tingkat Kepuasan
5	Sangat Setuju
4	Setuju
3	Cukup
2	Tidak Setuju
1	Sangat Tidak Setuju

2.2.10 Model Pengembangan Sistem *Waterfall*

Model *waterfall* merupakan model yang melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing* atau *verification*, dan *maintenance*. Secara umum gambaran tahapan pada model *waterfall* dapat dilihat pada Gambar 2.1.



Gambar 2.4 Model pengembangan sistem *waterfall*.

Penjelasan dari tahap-tahap yang dilakukan di dalam metode *waterfall* ini adalah sebagai berikut:

1. *Requirements analysis and definition*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software* untuk mengetahui sifat dari program yang akan dibuat.

2. *System and software design*

Proses yang digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk "*blueprint*" *software*. Desain harus dapat mengimplementasikan kebutuhan yang disebutkan pada tahap sebelumnya.

3. *Implementation and unit testing*

Proses untuk mengubah desain menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap desain yang secara teknis dikerjakan oleh *programmer*.

4. *Integration and system testing*

Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

5. *Operating and maintenance*

Pada tahapan ini dilakukan pemeliharaan terhadap *software*, perbaiki jika terjadi *error* kecil yang tidak ditemukan sebelumnya, dan juga menambahkan fitur-fitur yang belum ada pada *software* tersebut [10]

BAB III

METODE PENELITIAN

3.1 Alat dan Bahan

Kebutuhan pembuatan aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan ini membutuhkan beberapa alat dan bahan yang terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*) sebagai alat dan data-data penunjang pembuatan aplikasi sebagai bahan.

Perangkat keras yang dibutuhkan untuk membuat aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan adalah sebagai berikut:

1. Laptop dengan spesifikasi *processor* minimal Intel QuodCore dan RAM minimal 2 GB yang berfungsi sebagai media pembangun aplikasi *web* maupun aplikasi *mobile* dan di untuk keperluan uji coba sistem yang akan digunakan oleh *admin* dinas perhubungan dan juga Bina Marga di Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.
2. *Smartphone* sebagai media untuk uji coba sistem yang akan digunakan oleh *mobile user* dalam hal ini masyarakat Kota Mataram.

Perangkat lunak yang dibutuhkan untuk membuat aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan adalah sebagai berikut:

1. OS (*Operating System*), pada pembuatan aplikasi digunakan *Windows 7* sebagai media untuk menjalankan *software-software* yang akan digunakan untuk membuat sistem.
2. *Sublime Text* merupakan *text editor* yang akan digunakan untuk membuat *code* program dari aplikasi.
3. XAMPP merupakan *software web server* apache. Dalam penelitian ini akan dimanfaatkan *server* MySQL sebagai *server* yang digunakan untuk mengelola *database* aplikasi.

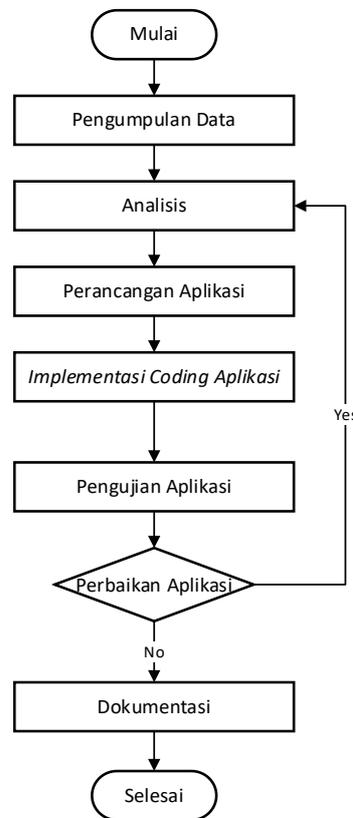
Bahan yang digunakan dalam pembangunan aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram antara lain:

1. Data jalan kota yang berupa karakteristik ruas jalan yaitu dari nama ruas jalan, panjang ruas, lebar, jenis permukaan, dan lain-lain pada tahun 2017 dari Bina Marga di Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram dalam bentuk *file Microsoft Excel*.

2. Data fasilitas kelengkapan jalan yang terpasang pada tahun 2014 sampai 2016 di Kota Mataram dari Dinas Perhubungan Kota Mataram dalam bentuk buku laporan.

3.2 Metode Pembuatan Aplikasi

Aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan studi kasus di Kota Mataram menggunakan metode pembuatan aplikasi dengan 6 tahap, yaitu pengumpulan data, analisis, perancangan aplikasi, implementasi/*coding* aplikasi, pengujian aplikasi, serta tahap terakhir adalah dokumentasi. Tahap-tahap pembuatan aplikasi tersebut diilustrasikan dalam diagram alir pada Gambar 3.1.



Gambar 3.1 Model pembuatan aplikasi.

3.2.1 Pengumpulan Data

Pada tahap pengumpulan data yang dibutuhkan untuk merancang dan membangun sistem, peneliti menggunakan teknik pengumpulan data dengan cara sebagai berikut :

3.2.1.1 Studi Literatur

Tahap studi literatur dilakukan dengan mempelajari dan memahami literatur seperti buku-buku, jurnal, serta mengakses beberapa situs terkait dengan penelitian yang dilakukan sehingga dapat menjadi sebagai pembanding dalam pembuatan aplikasi, dan

kekurangan yang terdapat pada penelitian terkait dapat dijadikan sebagai masukan dalam aplikasi yang akan dibuat.

3.2.1.2 Observasi

Tahap observasi dilakukan di Dinas Perhubungan Kota Mataram dan juga Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram, dimana hal tersebut bertujuan untuk mengetahui sistem yang sedang berjalan saat ini dan untuk mendapatkan data-data lengkap tentang informasi fasilitas kelengkapan jalan yang terpasang dan jalan kota di Kota Mataram, beserta bentuk laporan tahunan yang dilakukan oleh lembaga terkait jalan maupun kelengkapan jalan.

3.2.2 Analisis

Tahap analisis dilakukan dalam penelitian ini dibagi menjadi beberapa jenis analisis yaitu analisis sistem yang sedang berjalan dan analisis sistem yang akan dibangun.

3.2.2.1 Analisis sistem yang sedang berjalan

Pada tahap ini dilakukan analisis sistem proses pengelolaan data serta laporan yang dilakukan Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram untuk masalah jalan dan Dinas Perhubungan Kota Mataram untuk masalah fasilitas kelengkapan jalan. Setiap tahun Bina Marga melakukan pendataan ulang terhadap ruas jalan untuk melihat perubahan panjang dan presentasi dari setiap kategori kondisi jalan. Begitu pula pada Dinas Perhubungan, setiap tahunnya membuat pembukuan yang berisi pendataan pengadaan setiap kategori fasilitas kelengkapan yang dilakukan pada tahun tersebut dan laporan berupa tabel dan grafik peningkatan jumlah fasilitas kelengkapan yang terpasang setiap tahunnya. Proses pendataan yang dilakukan untuk jalan maupun fasilitas kelengkapan jalan belum memanfaatkan sebuah sistem yang saling berintegrasi, dimana data-data dalam setahun tersimpan dalam sebuah *file* berformat *document*. Hal tersebut dapat mengakibatkan kemungkinan kehilangan data yang sangat besar dan juga kesulitan dalam menganalisis kesimpulan data untuk membuat laporan tahunan.

Oleh karena itu, dalam penelitian ini akan dibuat sebuah sistem yang memudahkan dalam mengelola data yang sangat banyak dari setiap ruas jalan beserta kelengkapan jalan yang terpasang. Selain itu, sistem memberi kemudahan bagi masyarakat untuk mendapatkan informasi tentang kondisi jalan dan kelengkapan jalan perihal kerusakan, pengadaan dan juga perbaikan yang akan dilakukan.

3.2.2.2 Analisis sistem yang akan dibangun

Pada tahap ini dilakukan analisis sistem yang akan dibuat yaitu sistem pengolahan data yang memudahkan pemerintah dan masyarakat Kota Mataram untuk memantau kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram.

Pada sistem yang akan digunakan oleh bidang pada Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram dan Dinas Perhubungan Kota Mataram sebagai *admin* untuk melakukan pendataan juga memiliki fitur laporan, dimana fitur ini untuk melihat perkembangan statistik dari kondisi jalan maupun kelengkapan jalan berdasarkan masukan yang dilakukan oleh *admin* serta berdasarkan pengaduan kerusakan yang dilakukan masyarakat.

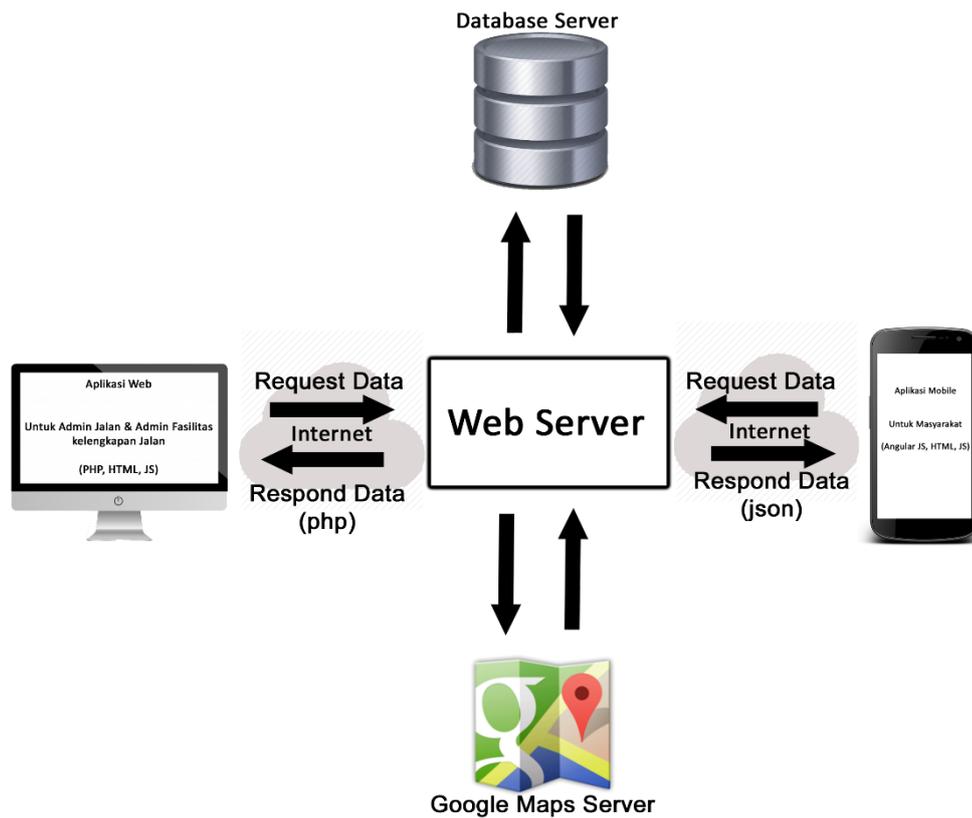
Pada sistem yang akan digunakan masyarakat memiliki fitur untuk melihat kondisi dan juga fitur pengaduan kerusakan jalan maupun kelengkapan jalan. Fitur melihat kondisi ini berdasarkan masukan yang dilakukan oleh *admin* dan pengaduan kerusakan yang telah dilakukan masyarakat lainnya. Sedangkan untuk fitur pengaduan kerusakan, pengaduan yang dilakukan langsung terkirim ke sistem yang digunakan oleh *admin* yang bertanggung jawab sesuai kategori pengaduan (jalan atau kelengkapan jalan).

3.2.3 Perancangan aplikasi

Pada tahap ini dilakukan perancangan aplikasi yang akan dibuat terdiri dari perancangan arsitektur aplikasi, diagram *use case*, *class*, *sequence*, *activity*, *entity relationship diagram*, dan desain *interface*.

3.2.3.1 Perancangan arsitektur aplikasi

Perancangan arsitektur aplikasi menggambarkan proses mengakses sistem yang mana aplikasi yang akan dibuat adalah *hybrid application* sehingga dapat di-*install* pada perangkat bergerak dengan sistem operasi *android* yang digunakan untuk masyarakat Kota Mataram. Selain itu, akan dibangun aplikasi *web* sebagai pelengkap *hybrid application* yang hanya dapat di akses oleh *admin*. Rancangan arsitektur aplikasi yang akan dibuat dapat dilihat pada Gambar 3.2.

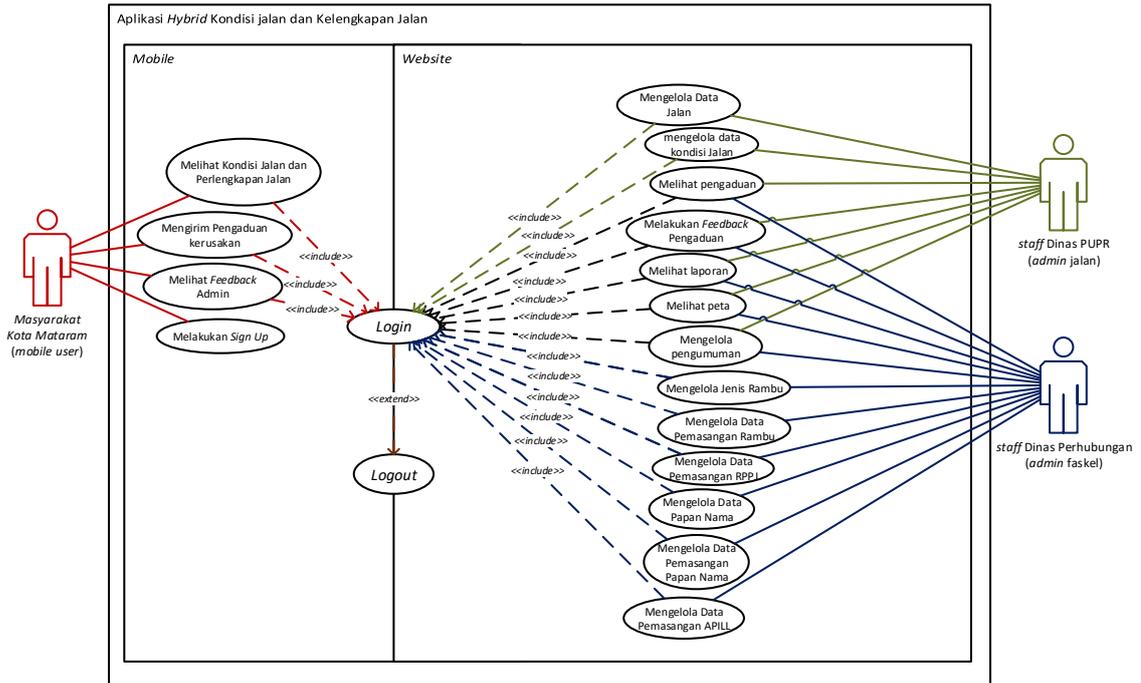


Gambar 3.2 Rancangan arsitektur aplikasi.

Pada Gambar 3.2 merupakan rancangan arsitektur aplikasi yang menggambarkan *user* yaitu masyarakat Kota Mataram pengguna aplikasi *mobile* maupun *admin* pengguna aplikasi *web* untuk mendapatkan data pada *Database Server* ataupun data dari *Google Maps Server*, maka sistem yang terhubung dengan internet akan melalukan *request* data ke *web server*. Sehingga kemudian *web server* akan memberikan respon data sesuai dengan yang diinginkan. Respon data dari *web server* untuk aplikasi *web* diberikan dalam bentuk *php*, sedangkan respon data untuk aplikasi *mobile* diberikan dalam bentuk *json*.

3.2.3.2 Diagram use case

Perancangan *use case* diagram untuk aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram bertujuan untuk mendeskripsikan fitur-fitur yang bisa dilakukan oleh *admin* jalan, *admin* fasilitas kelengkapan jalan, dan masyarakat Kota Mataram. *Use case* diagram dari aplikasi tersebut dapat dilihat pada Gambar 3.3.



Gambar 3.3 Use case diagram aplikasi hybrid.

Use case diagram menggambarkan proses yang dilakukan oleh actor atau pengguna sistem. Pada sistem yang akan dibangun terdapat tiga actor yaitu:

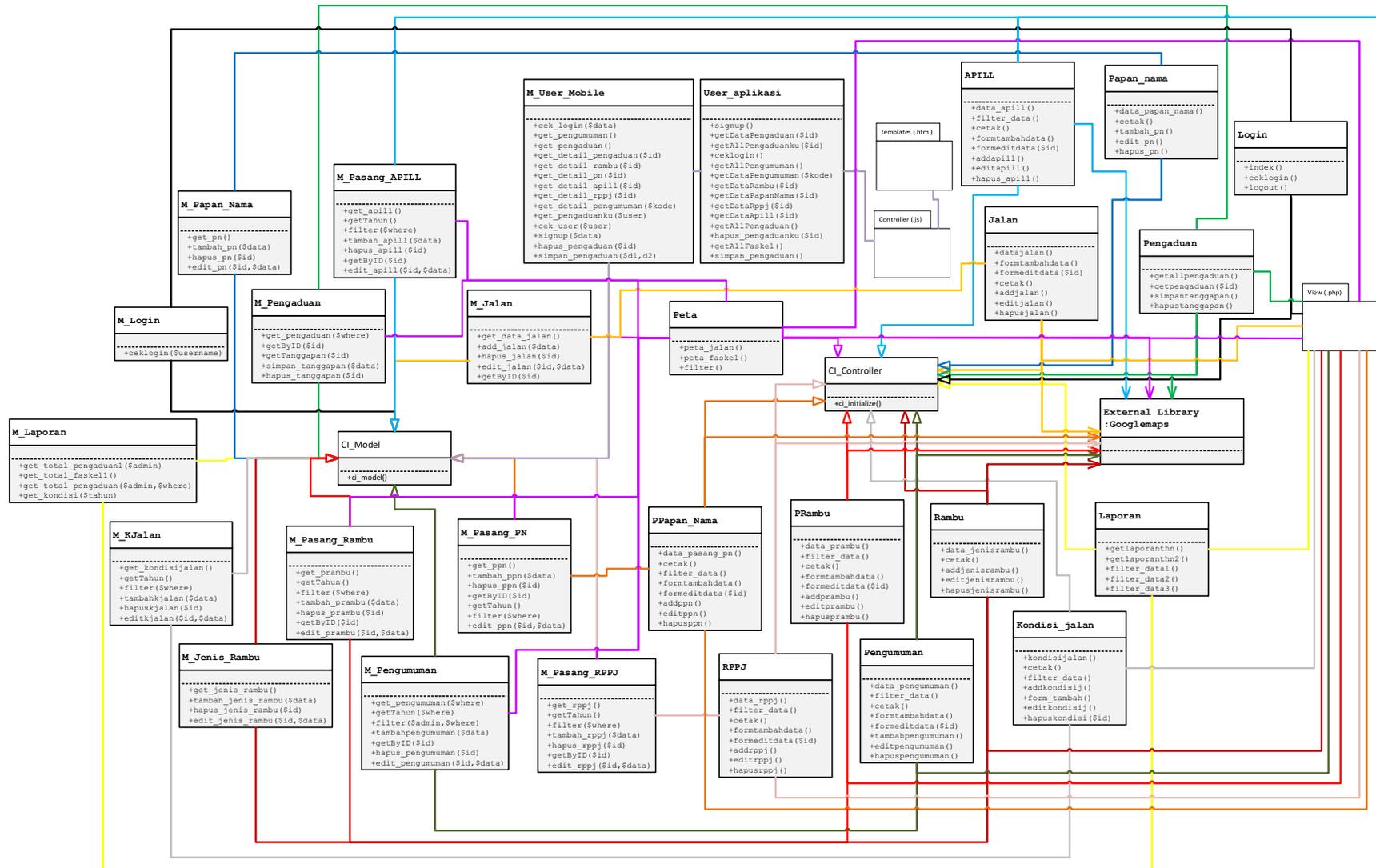
1. *Staff* Dinas Pekerjaan Umum dan Penataan Ruang (sebagai *admin* jalan)
 - a. Proses *login* untuk masuk dalam sistem.
 - b. Proses melihat peta setelah berhasil masuk ke sistem.
 - c. Proses mengelola pengumuman untuk menambahkan dan meng-*edit* pengumuman yang nanti dapat dilihat oleh *user*.
 - d. Proses mengelola data jalan untuk menambahkan data baru, mengubah data, dan menghapus data jalan.
 - e. Proses mengelola data kondisi jalan untuk menambah kondisi jalan terbaru, meng-*edit*, dan menghapus kondisi jalan.
 - f. Proses melihat pengaduan untuk melihat pengaduan yang telah dilakukan oleh masyarakat Kota Mataram.
 - g. Proses melihat laporan untuk melihat laporan akhir berdasarkan pendataan *admin* dan juga pengaduan yang pernah dilakukan.
 - h. Proses melakukan *feedback* pengaduan untuk mengirim *feedback* dari pengaduan yang dilakukan masyarakat berupa pesan.
 - i. Proses *logout* untuk keluar dari sistem.

2. *Staff Dinas Perhubungan (sebagai admin faskel)*
 - a. Proses *login* untuk masuk dalam sistem.
 - b. Proses melihat peta setelah berhasil masuk ke sistem.
 - c. Proses mengelola pengumuman untuk menambahkan dan meng-*edit* pengumuman yang nanti dapat dilihat oleh *user*.
 - d. Proses mengelola papan nama jalan untuk menambah gambar papan nama jalan baru, meng-*edit*, dan menghapus papan nama jalan.
 - e. Proses mengelola data pemasangan papan nama untuk menambah data baru, meng-*edit* data, dan menghapus data pemasangan papan nama jalan.
 - f. Proses mengelola jenis rambu untuk menambah jenis rambu baru, meng-*edit*, dan menghapus jenis rambu.
 - g. Proses mengelola data pemasangan rambu untuk menambah data baru, meng-*edit* data, dan menghapus data pemasangan rambu.
 - h. Proses mengelola data pemasangan APILL untuk menambah data baru, meng-*edit* data, dan menghapus data pemasangan APILL.
 - i. Proses mengelola data pemasangan RPPJ untuk menambah data baru, meng-*edit* data, dan menghapus data pemasangan RPPJ.
 - j. Proses melihat pengaduan untuk melihat pengaduan kerusakan fasilitas kelengkapan jalan yang telah dilakukan oleh masyarakat Kota Mataram.
 - k. Proses melakukan *feedback* pengaduan untuk mengirim *feedback* dari pengaduan kerusakan fasilitas kelengkapan jalan yang dilakukan masyarakat berupa pesan.
 - l. Proses melihat laporan untuk melihat laporan akhir berdasarkan pendataan *admin* dan juga pengaduan yang pernah dilakukan.
 - m. Proses *logout* untuk keluar dari sistem
3. Masyarakat Kota Mataram (sebagai *mobile user*)
 - a. Proses *login* untuk masuk dalam sistem.
 - b. Proses melakukan *sign up* untuk mendaftar akun baru.
 - c. Proses mengirim pengaduan untuk melakukan pengaduan kerusakan jalan maupun kerusakan fasilitas kelengkapan jalan.
 - d. Proses melihat *feedback admin* untuk melihat *feedback* atau tanggapan dari *admin* berupa pesan terhadap pengaduan yang telah dilakukan sebelumnya.

- e. Proses melihat kondisi jalan dan perlengkapan jalan untuk melihat kondisi jalan dan perlengkapan jalan yang terpasang di Kota Mataram berdasarkan pengaduan-pengaduan kerusakan yang telah dilakukan oleh masyarakat Kota Mataram, pendataan fasilitas kelengkapan dan info dari *admin* jika akan ada perbaikan yang dilakukan.
- f. Proses *logout* untuk keluar dari sistem.

3.2.3.3 Diagram *class*

Class diagram menggambarkan keadaan dari sistem yang akan dibangun terdiri dari beberapa *class* yang berisi atribut dan *method*, serta hubungan masing-masing *class*. Perancangan *class* diagram aplikasi ini dapat dilihat pada Gambar 3.4.



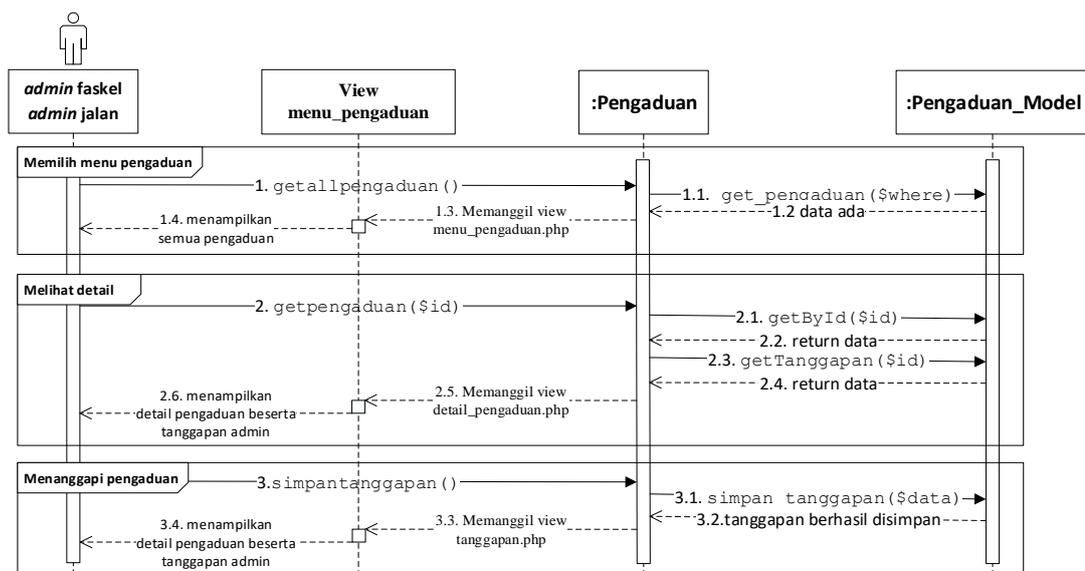
Gambar 3.4 Class diagram sistem.

Pada Gambar 3.4 yang merupakan *class diagram* dari sistem, dimana setiap *class* saling terhubung. *View* merupakan kumpulan *source code* yang akan memberikan tampilan di dalam sistem. *Class controller* yaitu *Class* Login, APILL, RPPJ, Papan_Nama, PPapan_Nama Rambu, PRambu, Kondisi_jalan, Jalan, Pengumuman, Pengaduan, Peta, Laporan, dan User_aplikasi merupakan turunan dari *class* CI_Controller yang digunakan untuk menghubungkan *class view* dan *class* model. Sedangkan *class* model yaitu *class* M_Mobile_User, M_Jalan, M_Jenis_Rambu, M_KJalan, M_Login, M_Pasang_APILL, M_Papan_Nama, M_Pasang_PN, M_Pasang_Rambu, M_Pasang_RPPJ, M_Pengaduan, M_Pengumuman, dan M_Laporan merupakan turunan dari kelas CI_Model yang digunakan dalam proses manipulasi data di *database*. Beberapa *class controller* pada sistem menggunakan fungsi-fungsi yang ada pada *library* Googlemap.

3.2.3.4 Diagram *sequence*

Sequence diagram menggambarkan tahapan dari proses tertentu. *Sequence* diagram dari aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram dirancang berdasarkan *use case diagram* pada Gambar 3.3. Adapun *sequence diagram* secara lengkap dapat dilihat pada lampiran I. Berikut adalah beberapa rancangan *sequence diagram* untuk melakukan respon dan permintaan data oleh *user* dan *admin*.

1. *Sequence diagram* proses melihat pengaduan dan melakukan *feedback*

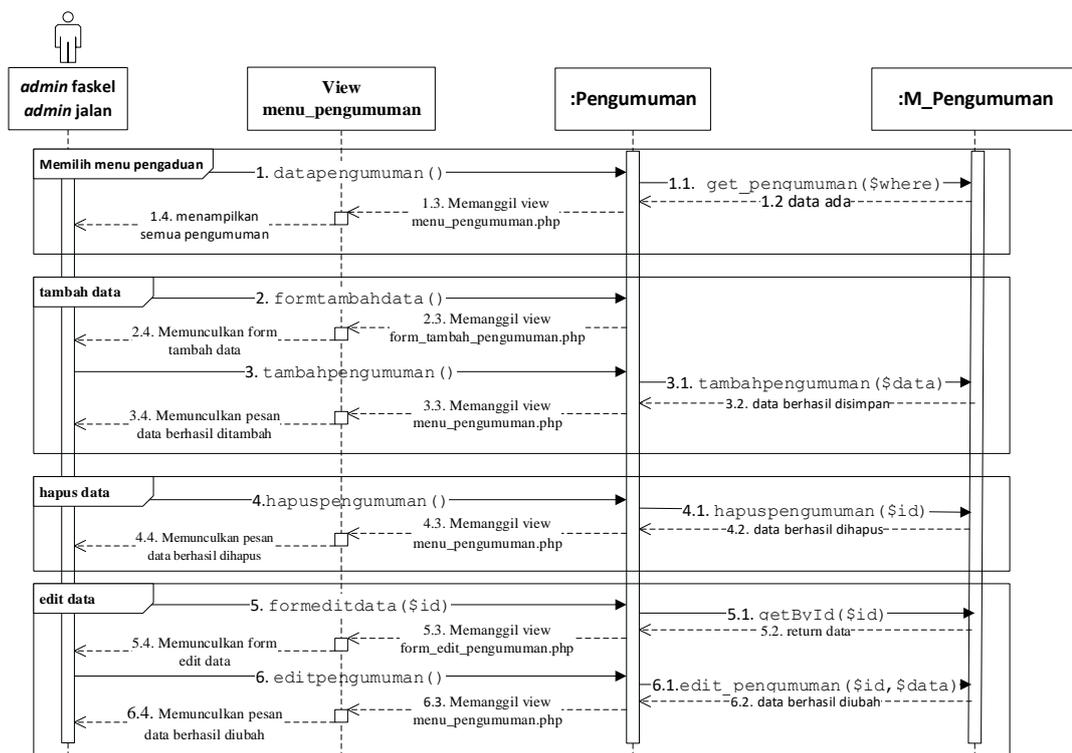


Gambar 3.5 *Sequence diagram* proses melihat pengaduan dan melakukan *feedback*.

Pada Gambar 3.5 merupakan *sequence diagram* untuk proses melihat pengaduan dan melakukan *feedback* oleh *actor* *admin* jalan dan *admin* faskel. Memilih menu

pengaduan, maka akan menjalankan fungsi `getallpengaduan()` pada *class* Pengaduan dan akan memanggil fungsi `get_pengaduan($where)` pada *class* M_Pengaduan untuk mengambil semua data pengaduan yang ada sesuai dengan kategori pengaduan dari *database* kemudian sistem akan menampilkan *list* data pengaduan yang dapat dilihat aktor. Untuk melihat detail satu pengaduan yang terdapat di tabel akan menjalankan fungsi `getpengaduan($id)` pada *class* Pengaduan dan juga fungsi `getByID($id)` dan `getTanggapan($id)` pada *class* M_Pengaduan untuk mengambil satu data pengaduan beserta tanggapan yang telah dilakukan sebelumnya jika ada dari *database* untuk ditampilkan secara detail. Jika ingin menambahkan *feedback*, aktor tinggal mengisi *form* dan menekan tombol *submit* yang akan menjalankan fungsi `simpantanggapan ($where)` pada *class* Pengaduan dan memanggil fungsi `simpan_tanggapan($data)` pada *class* M_Pengaduan untuk menyimpan tanggapan tersebut kedalam *database*.

2. Sequence diagram proses mengelola pengumuman



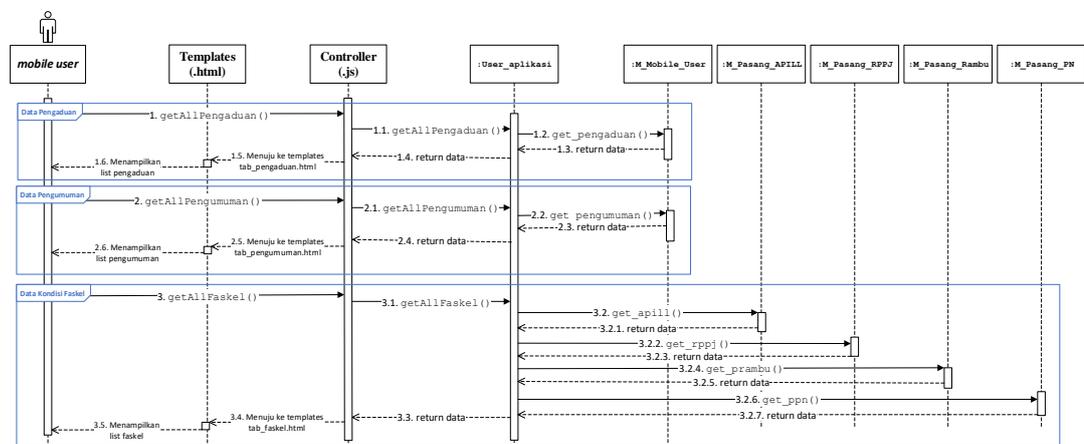
Gambar 3.6 Sequence diagram proses mengelola pengumuman.

Pada Gambar 3.6 merupakan *sequence* diagram dari proses mengelola pengumuman oleh *admin* jalan maupun *admin* faskel.

Untuk tambah data ketika aktor telah mengisi data baru pada *form* tambah data dan menekan tombol simpan maka akan menjalankan fungsi `tambahpengumuman ()` pada

class Pengumuman dan menjalankan fungsi `tambahpengumuman($data)` pada *class* *M_Pengumuman* untuk menyimpan data kedalam *database*. Untuk *edit* data ketika aktor telah mengisi pemperbaru data pada *form edit* data dan menekan tombol simpan maka akan menjalankan fungsi `editpengumuman()` pada *class* Pengumuman dan menjalankan fungsi `edit_pengumuman($id,$data)` pada *class* *M_Pengumuman* untuk meng-*update* data yang ada dalam *database*. Untuk menghapus data ketika aktor menekan tombol hapus maka akan menjalankan fungsi `hapuspengumuman()` pada *class* Pengumuman dan menjalankan fungsi `hapus_pengumuman($data)` pada *class* *M_Pengumuman* untuk menghapus data dari *database*.

3. Sequence diagram proses melihat kondisi jalan dan fasilitas kelengkapan

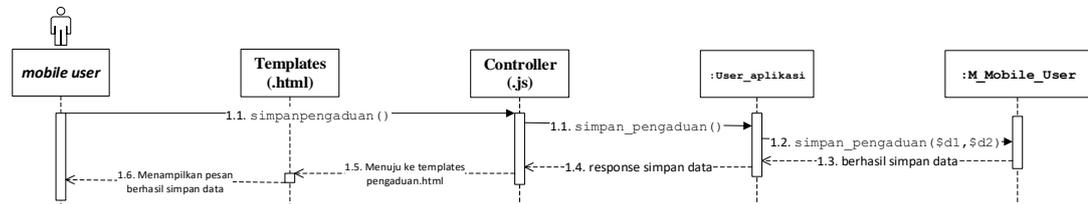


Gambar 3.7 Sequence diagram proses melihat kondisi jalan dan kelengkapan jalan.

Pada Gambar 3.7 merupakan *sequence* diagram proses melihat kondisi jalan dan kelengkapan jalan, dimana proses ini merupakan menu *home* pada aplikasi *mobile* saat aktor telah berhasil *login*. Pada menu *home* terdapat tiga tab yang dapat diakses yaitu tab pengaduan, ketika dipilih maka akan memanggil fungsi `getAllPengaduan()` pada *class* *pengaduanCtrl* dan *request* data ke server dengan memanggil fungsi `getAllPengaduan()` pada *class* *User_aplikasi* selanjutnya mengambil data pada *database* dengan fungsi `get_pengaduan()` pada *class* *M_Mobile_User*. Tab pengumuman, menjalankan fungsi `getAllPengumuman()` pada *class* *pengumumanCtrl* dan *request* data ke server dengan memanggil fungsi `getAllPengumuman()` pada *class* *User_aplikasi* selanjutnya mengambil data pada *database* dengan fungsi `get_pengumuman()` pada *class* *M_Mobile_User*. Tab faskel, menjalankan fungsi `getAllFaskel()` pada *class* *pengaduanCtrl* dan *request* data ke server dengan memanggil fungsi `getAllFaskel()` pada *class* *User_aplikasi* selanjutnya mengambil data pada *database* dengan fungsi `get_apill()` pada *class*

M_Pasang_APILL, fungsi `get_rppj()` pada *class* M_Pasang_RPPJ, fungsi `get_prambu()` pada *class* M_Pasang_Rambu, dan fungsi `get_ppn()` pada *class* M_Pasang_PN.

4. Sequence diagram proses melakukan pengaduan



Gambar 3.8 Sequence diagram proses melakukan pengaduan.

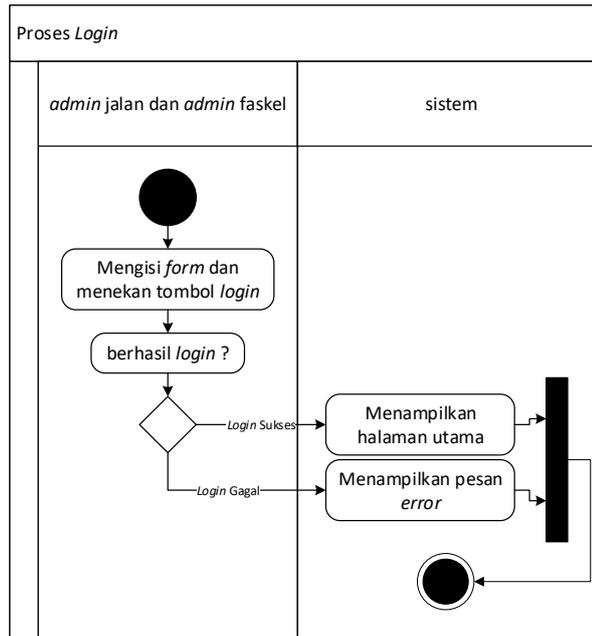
Pada Gambar 3.8 merupakan *sequence* diagram dari proses melakukan pengaduan, dimana ketika aktor memilih melakukan pengisian untuk *form* pengaduan kemudian menekan tombol Kirim maka akan menjalankan fungsi `simpanpengaduan()` pada *class* `laporjalanCtrl` maupun *class* `laporfaskelCtrl`, kemudian data dikirim ke server dengan menjalankan fungsi `simpan_pengaduan()` pada *class* `User_aplikasi` dan selanjutnya data disimpan pada *database server* dengan menjalankan fungsi `simpan_pengaduan($d1, $d2)` pada *class* `M_Mobile_User`.

3.2.3.5 Diagram activity

Activity diagram menggambarkan aktivitas-aktivitas atau aliran kerja dari sebuah sistem. Berikut ini adalah *Activity* diagram dari aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram.

1. Activity diagram proses login admin

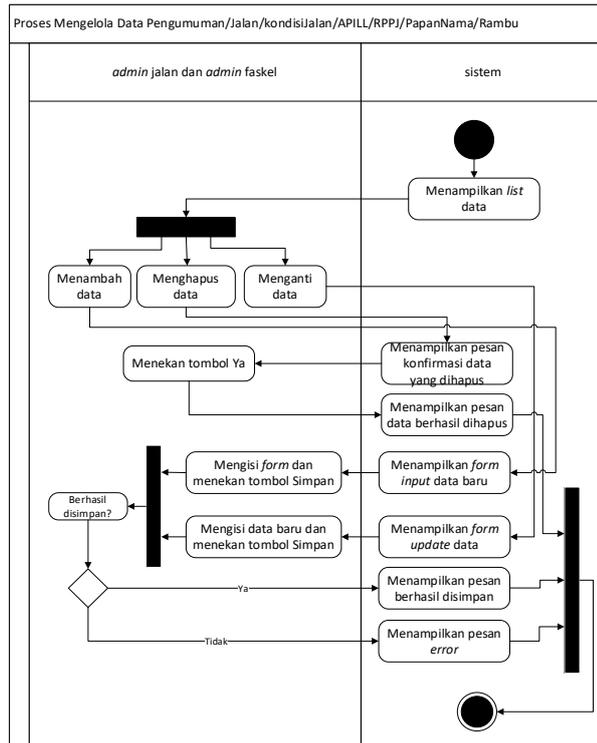
Gambar 3.9 merupakan *activity* diagram *login* ke sistem, untuk masuk ke sistem terdapat *username* dan *password* yang harus di-input ke *form login* setelah memilih menu *login*. Jika data ada di *database* maka akan ke halaman *home* dan beberapa menu dapat diakses dan jika data tidak ada di *database* maka akan menampilkan pesan *error* dan kembali ke *form login*.



Gambar 3.9 Activity diagram proses login admin.

2. Activity diagram proses mengelola data

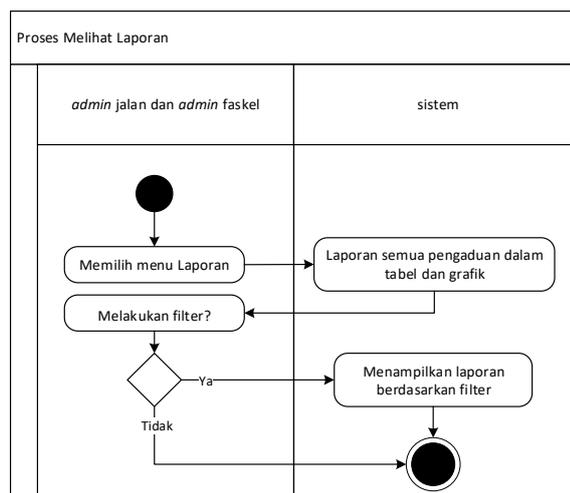
Gambar 3.10 merupakan *activity* diagram proses mengelola data jalan, data fasilitas kelengkapan, pengumuman, kondisi jalan, maupun jenis rambu, pengadaan papan nama, pemasangan papan nama, pemasangan RPPJ, pemasangan APILL, dan pemasangan rambu . Sistem akan menampilkan seluruh data kemudian ada tiga tindakan yang dapat dilakukan yaitu tambah data, hapus data, dan meng-*edit* data. Ketika memilih tombol hapus maka akan menampilkan pesan konfirmasi data yang akan dihapus kemudia menekan tombol Ya dan sistem akan memunculkan pesan berhasil dihapus jika memilih tambah data maka akan menampilkan *form* tambah data kemudia mengisi semua *field* begitu pula jika ingin meng-*edit* akan mengarahkan ke *form edit* data dan mengganti *field* yang ingin dirubah jika data baru ataupun data yang dirubah berhasil disimpan akan menampilkan pesan berhasil tapi jika tidak maka akan menampilkan pesan *error*.



Gambar 3.10 Activity diagram proses mengelola data.

3. Activity diagram proses melihat laporan

Gambar 3.11 merupakan *activity* diagram proses melihat laporan, *admin* dapat memilih menu laporan untuk melihat laporan tahunan dalam bentuk tabel dan grafik berdasarkan pengaduan yang telah dilakukan dan *admin* dapat mem-*filter* laporan yang akan dilihat.

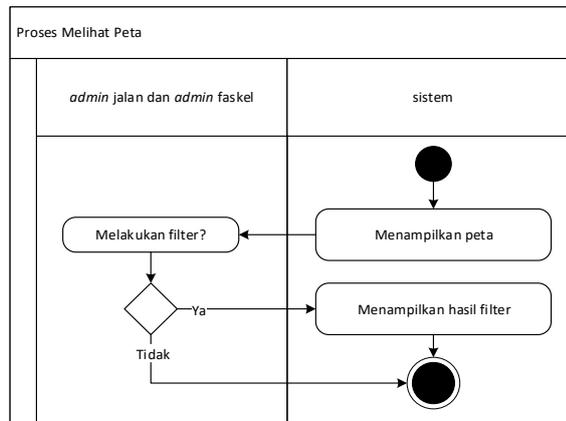


Gambar 3.11 Activity diagram proses melihat laporan.

4. Activity diagram proses melihat peta

Gambar 3.12 merupakan *activity* diagram proses melihat peta, *admin* setelah login akan diarahkan ke halaman utama yaitu menu peta untuk melihat data jalan maupun

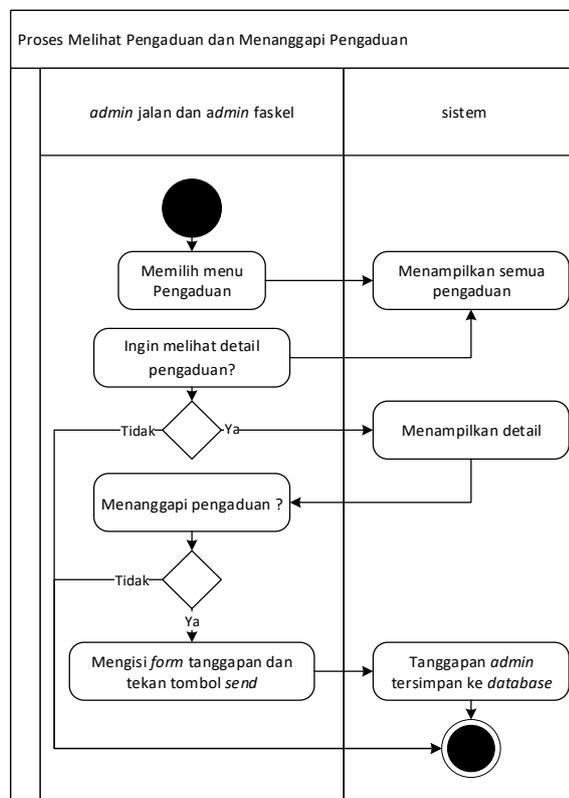
fasilitas kelengkapan jalan dalam bentuk peta dan *admin* dapat mem-*filter* data yang akan dimunculkan dalam peta.



Gambar 3.12 Activity diagram proses melihat peta.

5. Activity diagram proses melihat pengaduan

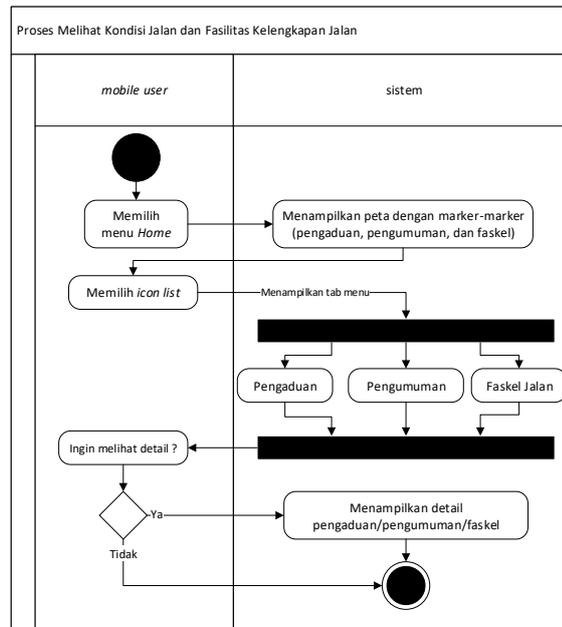
Gambar 3.13 merupakan *activity* diagram proses melihat pengaduan, untuk itu *admin* yang telah berhasil *login* bisa mengakses menu pengaduan maka sistem akan menampilkan semua pengaduan yang telah dilakukan oleh masyarakat dengan kategori berdasarkan *admin* yang aktif tersebut. Selain itu *admin* juga dapat memilih untuk melihat detail pengaduan serta melakukan tanggapan atas pengaduan tersebut.



Gambar 3.13 Activity diagram proses melihat pengaduan.

6. Activity diagram proses melihat kondisi

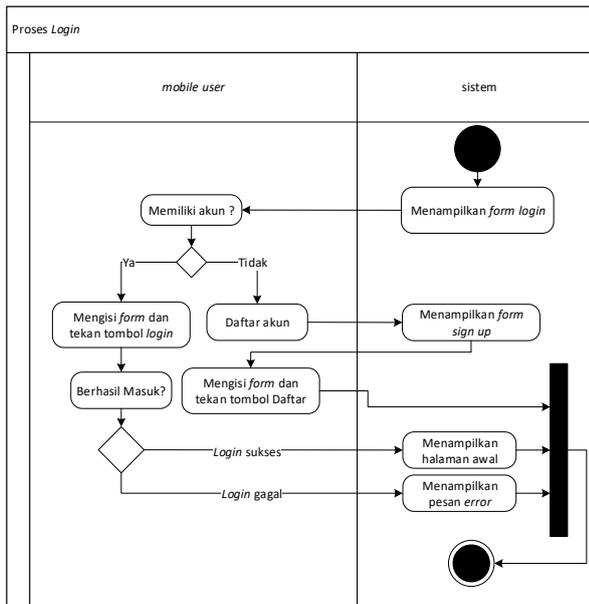
Gambar 3.14 merupakan *activity* diagram proses melihat kondisi, setelah *mobile user* berhasil *login* maka sistem akan mengarahkan langsung ke halaman utama dan menge-load sebuah peta dan marker-marker dari pengaduan yang telah dilakukan dan posisi dari fasilitas kelengkapan yang terpasang.



Gambar 3.14 Activity diagram proses melihat kondisi.

7. Activity diagram proses melakukan login dan signup user

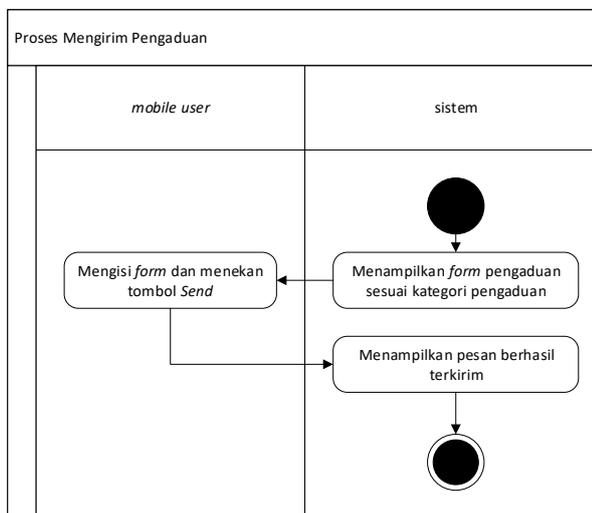
Gambar 3.15 merupakan *activity* diagram *login* dan *signup* ke sistem, untuk masuk ke sistem terdapat *username* dan *password* yang harus di-input ke *form login* setelah memilih menu *login*. Jika data ada di *database* maka akan ke halaman *home* dan beberapa menu dapat diakses dan jika data tidak ada di *database* maka akan menampilkan pesan *error* dan kembali ke *form login*. Jika belum memiliki akun maka dapat daftar akun dengan memilih link menuju *form sign up*, kemudian mengisi semua data identitas diri beserta mendaftarkan *username* dan *password* yang akan digunakan untuk masuk ke sistem.



Gambar 3.15 Activity diagram proses login dan signup user.

8. Activity diagram proses melakukan pengaduan

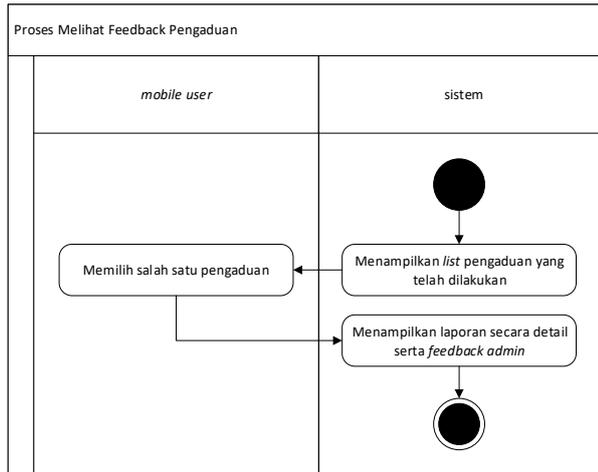
Gambar 3.16 merupakan activity diagram proses melakukan pengaduan, setelah *mobile user* berhasil login dan memilih menu Laporan. Sistem menampilkan form pengaduan berdasarkan kategori yang dipilih dan *mobile user* mengisi form dan menekan tombol *send*.



Gambar 3.16 Activity diagram proses melakukan pengaduan.

9. Activity diagram proses melihat feedback

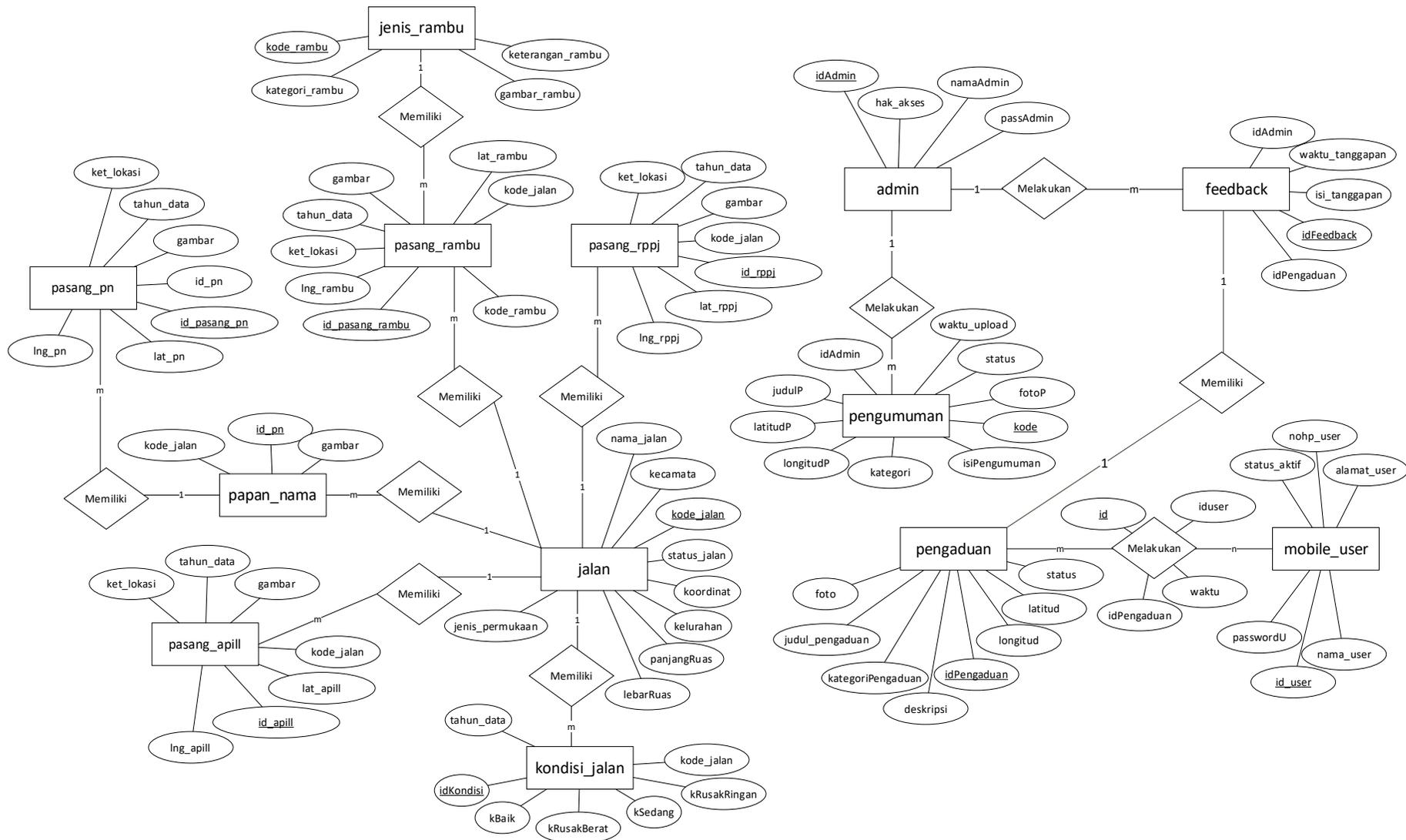
Gambar 3.17 merupakan activity diagram proses melihat feedback dari admin, *mobile user* dapat memilih menu pengaduanku untuk melihat list pengaduan yang telah dilakukannya, *user* memilih salah satu pengaduan kemudian akan ditampilkan detail pengaduan beserta feedback dari admin berupa pesan singkat.



Gambar 3.17 Activity diagram proses melihat *feedback*.

3.2.3.6 Entity relationship diagram

Entity relationship diagram (ERD) dalam aplikasi *hybrid* yang akan dibangun terdiri dari 13 entitas yaitu *admin*, pengumuman, masyarakat, pengaduan, kondisi_jalan, *feedback*, pasang_apill, pasang_rppj, papan_nama, pasang_pn, jenis_rambu, pasang_rambu, dan jalan. ERD aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan dapat dilihat pada Gambar 3.18.



Gambar 3.18 Entity relationship diagram aplikasi.

Pada Gambar 3.18 dapat dilihat bahwa setiap entitas didalamnya memiliki kaitan atau relasi dengan entitas lainnya, sehingga tidak ada satupun entitas yang berdiri sendiri. Struktur tabel dari ERD Aplikasi *hybrid* kondisi jalan dan fasilitas kelengkapan jalan di Kota Mataram dapat dijelaskan pada tabel berikut:

Tabel 3.1 Struktur tabel *admin*.

No.	Field	Tipe Data	Key	Keterangan
1	<u>idAdmin</u>	char(3)	Primary Key	Primary Key tabel admin. Sebagai <i>username</i> untuk <i>login</i> aplikasi
2	namaAdmin	varchar(15)		Nama <i>admin</i>
3	passAdmin	varchar(15)		Password melakukan <i>login</i>
4	hak_akses	enum('faskel kota', 'jalan kota')		Hak akses <i>admin</i> dalam sistem

Tabel 3.2 Struktur tabel *mobile_user*.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_user</u>	varchar(10)	Primary Key	Primary Key tabel masyarakat. Sebagai <i>username</i> (nama akun) untuk <i>login</i>
2	nama_user	varchar(30)		Nama <i>user</i> aplikasi
3	alamat_user	varchar(50)		Alamat <i>user</i> aplikasi
4	nohp_user	varchar(12)		Nomor HP <i>user</i> aplikasi
5	status_aktif	enum('Aktif', 'Tidak Aktif')		Status akun <i>user</i> aplikasi
6	passwordU	varchar(10)		Password <i>user</i> aplikasi

Tabel 3.3 Struktur tabel *jalan*.

No.	Field	Tipe Data	Key	Keterangan
1	<u>kode_jalan</u>	varchar(12)	Primary Key	Primary Key table jalan
2	nama_jalan	varchar(30)		Nama jalan
3	jenis_permukaan	varchar(20)		Jenis Permukaan jalan
4	panjangRuas	double		Panjang ruas jalan
5	lebarRuas	double		Lebar ruas jalan
6	kecamatan	varchar(20)		Kecamatan yang dilalui jalan
7	kelurahan	varchar(30)		Kelurahan yang dilalui jalan
8	status_jalan	enum('Jalan Kota', 'Jalan Nasional', 'Jalan Provinsi')		Status jalan
9	koordinat	text		Path koordinat jalan

Tabel 3.4 Struktur tabel *kondisi_jalan*.

No.	Field	Tipe Data	Key	Keterangan
1	<u>idKondisi</u>	varchar(10)	Primary Key	Primary Key table kondisi
2	kode_jalan	varchar(12)	Foreign Key	Foreign Key pada tabel kondisi
4	kBaik	double		Presentasi kondisi jalan baik

5	kSedang	double		Presentasi kondisi jalan sedang
6	kRusakRingan	double		Presentasi kondisi jalan rusak ringan
7	kRusakBerat	double		Presentasi kondisi jalan rusak berat
9	tahun_data	year(4)		Tahun data kondisi jalan

Tabel 3.5 Struktur tabel jenis_rambu.

No.	Field	Tipe Data	Key	Keterangan
1	<u>kode_rambu</u>	varchar(12)	Primary Key	Primary Key tabel jenis_rambu
2	kategori_rambu	enum('Larangan', 'Petunjuk', 'Peringatan', 'Perintah')		Kategori dari jenis rambu
3	keterangan_rambu	text		Keterangan jenis rambu
4	gambar_rambu	varchar(20)		Gambar jenis rambu

Tabel 3.6 Struktur tabel pasang_rambu.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_pasang_rambu</u>	varchar(15)	Primary Key	Primary Key tabel pasang_rambu
2	kode_rambu	varchar(12)	Foreign Key	Foreign Key dari tabel pasang_rambu
3	kode_jalan	varchar(12)	Foreign Key	Foreign Key dari tabel pasang_rambu
4	lat_rambu	double		Latitude rambu dipasang
5	lng_rambu	double		Longitude rambu dipasang
6	ket_lokasi	varchar(30)		Keterangan lokasi pemasangan
7	tahun_data	year(4)		Tahun data pemasangan
8	gambar	varchar(20)		Foto kondisi pemasangan

Tabel 3.7 Struktur tabel papan_nama.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_pn</u>	varchar(15)	Primary Key	Primary Key tabel papan_nama
2	kode_jalan	varchar(12)	Foreign Key	Foreign Key dari tabel papan_nama
3	gambar	varchar(15)		Gambar Papan Nama

Tabel 3.8 Struktur tabel pasang_pn.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_pasang_pn</u>	varchar(15)	Primary Key	Primary Key tabel pasang_pn
2	id_pn	varchar(15)	Foreign Key	Foreign Key dari tabel pasang_pn
3	lat_pn	double		Latitude papan nama dipasang
4	lng_pn	double		Longitude papan nama dipasang
5	ket_lokasi	varchar(30)		Keterangan lokasi pemasangan
6	tahun_data	year(4)		Tahun data pemasangan
7	gambar	varchar(20)		Foto kondisi pemasangan

Tabel 3.9 Struktur tabel pasang_apill.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_apill</u>	varchar(15)	Primary Key	Primary Key tabel pasang_apill
2	kode_jalan	varchar(12)	Foreign Key	Foreign Key dari tabel pasang_apill
3	lat_apill	Double		Latitude apill dipasang
4	lng_apill	Double		Longitude apill dipasang
5	ket_lokasi	varchar(30)		Keterangan lokasi pemasangan
6	tahun_data	year(4)		Tahun data pemasangan
7	gambar	varchar(20)		Foto kondisi pemasangan

Tabel 3.10 Struktur tabel pasang_rppj.

No.	Field	Tipe Data	Key	Keterangan
1	<u>id_rppj</u>	varchar(15)	Primary Key	Primary Key tabel pasang_rppj
2	kode_jalan	varchar(12)	Foreign Key	Foreign Key dari tabel pasang_rppj
3	lat_rppj	Double		Latitude rppj dipasang
4	lng_rppj	Double		Longitude rppj dipasang
5	ket_lokasi	varchar(30)		Keterangan lokasi pemasangan
6	tahun_data	year(4)		Tahun data pemasangan
7	gambar	varchar(20)		Foto kondisi pemasangan

Tabel 3.11 Struktur tabel pengumuman

No.	Field	Tipe Data	Key	Keterangan
1	<u>kode</u>	varchar(15)	Primary Key	Primary Key tabel tanggapan
2	idAdmin	varchar(10)	Foreign Key	Foreign Key dari tabel admin
3	judulP	varchar(50)		Judul pengumuman
4	kategori	varchar(10)		Kategori pengumuman
5	isiPengumuman	text		Isi pengumuman
6	waktu_upload	datetime		Waktu upload pengumuman
7	status	enum('Aktif', 'Sembunyikan')		Status pengumuman untuk menentukan akan ditampilkan atau tidak
8	fotoP	varchar(20)		Foto sebagai lampira
9	latitudP	double		Latitude dari posisi dilakukan pembanguna atau perbaikan
10	longitudP	double		Longitude dari posisi dilakukan pembanguna atau perbaikan

Tabel 3.12 Struktur tabel pengaduan.

No.	Field	Tipe Data	Key	Keterangan
1	<u>idPengaduan</u>	varchar(10)	Primary Key	Primary Key tabel pengaduan
2	kategoriPengaduan	varchar(25)		Kategori dari pengaduan yang dilakukan
3	judul_pengaduan	varchar(25)		Judul pengaduan
4	deskripsi	text		Deskripsi kerusakan
5	status	enum('Belum dibaca',		Status pengaduan

		'Tampilkan', 'Sembunyikan')		
6	latitud	double		<i>Latitude</i> dari posisi masyarakat saat melakukan pengaduan
7	longitud	double		<i>Longitude</i> dari posisi masyarakat saat melakukan pengaduan
8	foto	varchar(25)		Foto untuk bukti pengaduan kerusakan

Tabel 3.13 Struktur tabel *feedback*.

No.	<i>Field</i>	Tipe Data	<i>Key</i>	Keterangan
1	<u>idFeedback</u>	varchar(15)	<i>Primary Key</i>	<i>Primary Key</i> tabel tanggapan dengan <i>Auto_Increment</i>
2	idAdmin	varchar(10)	<i>Foreign Key</i>	<i>Foreign Key</i> pada tabel tanggapan
3	idPengaduan	varchar(10)	<i>Foreign Key</i>	<i>Foreign Key</i> pada tabel tanggapan
4	isi_tanggapan	text		Isi tanggapan dari <i>admin</i>
5	waktu_tanggapan	datetime		Waktu tanggapan dilakukan

Tabel 3.14 Struktur tabel melakukan pengaduan.

No.	<i>Field</i>	Tipe Data	<i>Key</i>	Keterangan
1	<u>Id</u>	varchar(10)	<i>Primary Key</i>	<i>Primary Key</i> relasi melakukan
2	Iduser	varchar(10)	<i>Foreign Key</i>	<i>Foreign Key</i> pada tabel <i>mobile_user</i>
3	idPengaduan	varchar(10)	<i>Foreign Key</i>	<i>Foreign Key</i> pada tabel pengaduan
4	Waktu	datetime		Waktu melakukan pengaduan

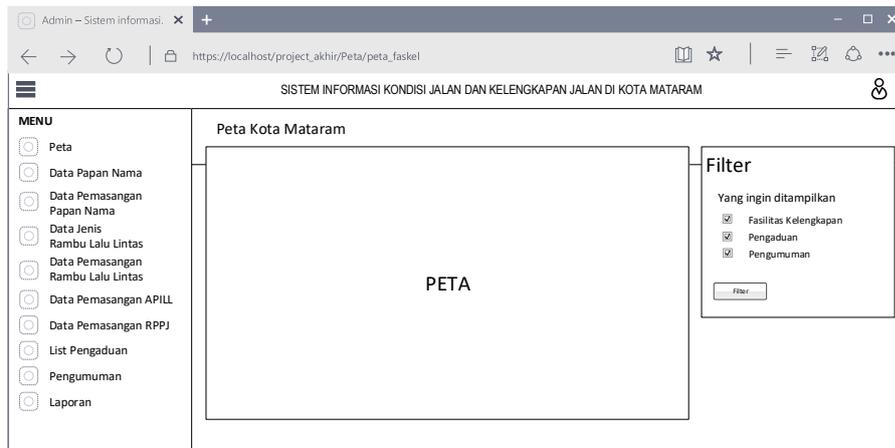
3.2.3.7 Perancangan Tampilan

Perancangan tampilan aplikasi merupakan perancangan antar muka dari aplikasi yang akan digunakan oleh *admin* dan masyarakat Kota Mataram.

3.2.3.7.1 Rancangan tampilan aplikasi *web* untuk *admin* faskel

Perancangan tampilan aplikasi *web* yang akan digunakan oleh *admin* faskel atau Dinas Perhubungan Kota Mataram terdiri dari beberapa tampilan sebagai berikut:

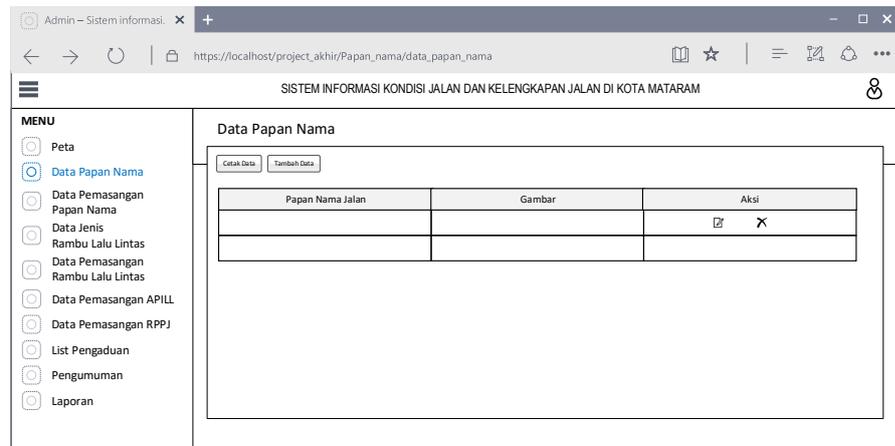
1. Tampilan halaman menu peta untuk admin faskel



Gambar 3.19 Tampilan halaman menu peta untuk admin faskel.

Gambar 3.19 merupakan menu utama untuk admin faskel setelah berhasil *login*. Menampilkan sebuah peta dan juga koordinat yang berkaitan dengan fasilitas kelengkapan jalan seperti koordinat faskel yang terpasang, yang rusak, dan yang akan diperbaiki atau dipasang.

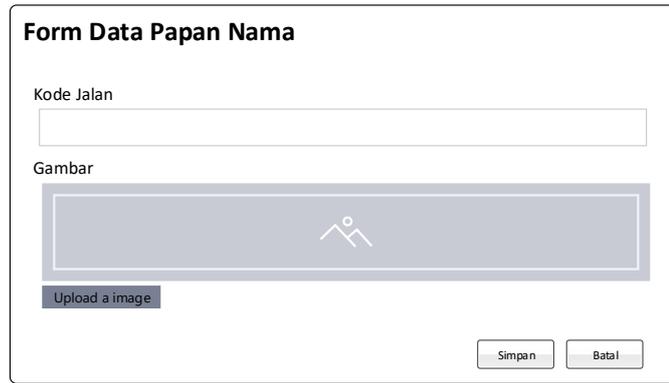
2. Tampilan halaman menu data papan nama



Gambar 3.20 Tampilan halaman menu data papan nama.

Gambar 3.20 merupakan halaman menu data papan nama yang digunakan oleh *admin* fasilitas kelengkapan. Pada halaman ini menampilkan tabel untuk pengadaan papan nama. Ada 3 proses yang dapat dilakukan dalam halaman ini, yaitu tambah data baru, *edit* data, dan hapus data.

3. Tampilan halaman *form* data papan nama



Form Data Papan Nama

Kode Jalan

Gambar

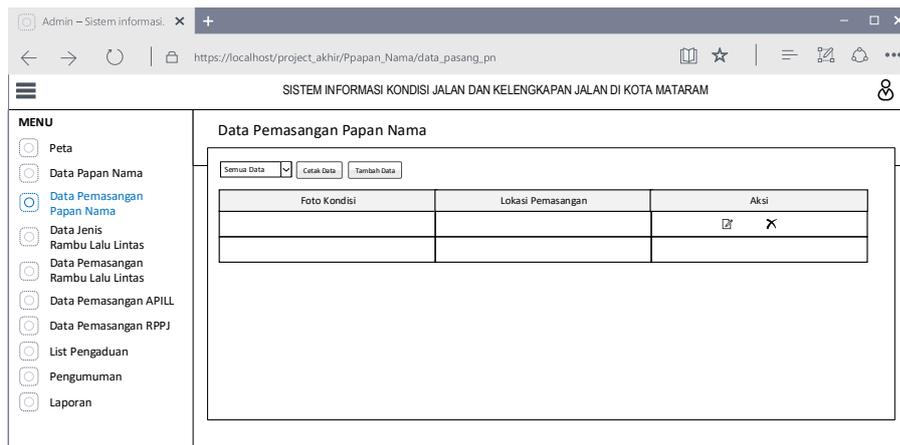
Upload a image

Simpan Batal

Gambar 3.21 Tampilan halaman *form* data papan nama.

Gambar 3.21 merupakan halaman *form* papan nama yang digunakan oleh *admin* fasilitas kelengkapan. Pada halaman ini menampilkan *form* untuk menambahkan atau meng-*edit* data.

4. Tampilan halaman menu data pemasangan papan nama



Admin – Sistem informasi. x +

https://localhost/project_akhir/Ppapan_Nama/data_pasang_pn

SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Papan Nama
- Data Pemasangan Papan Nama**
- Data Jenis
- Rambu Lalu Lintas
- Data Pemasangan Rambu Lalu Lintas
- Data Pemasangan APILL
- Data Pemasangan RPPJ
- List Pengaduan
- Pengumuman
- Laporan

Data Pemasangan Papan Nama

Semua Data Cetak Data Tambah Data

Foto Kondisi	Lokasi Pemasangan	Aksi
		 

Gambar 3.22 Tampilan halaman menu data pemasangan papan nama.

Gambar 3.22 merupakan halaman menu data pemasangan papan nama yang digunakan oleh *admin* fasilitas kelengkapan, pada halaman ini ditampilkan semua data pemasangan yang dilakukan dalam tabel. Pada setiap data yang ada dapat dilakukan dua tindakan yaitu menghapus data, dan merubah data.

5. Tampilan halaman *form* pemasangan papan nama

Admin – Sistem Informasi. x +
https://localhost/project_akhir/Ppapan_Nama/
SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Papan Nama
- Data Pemasangan Papan Nama
- Data Jenis Rambu Lalu Lintas
- Data Pemasangan Rambu Lalu Lintas
- Data Pemasangan APILL
- Data Pemasangan RPPJ
- List Pengaduan
- Pengumuman
- Laporan

Form Data Pemasangan Papan Nama

Tahun Data:

Nama Jalan:

Keterangan Lokasi:

Lokasi Pada Peta:

Latitud: Longitudud:

Gambar:

Gambar 3.23 Tampilan halaman *form* pemasangan papan nama.

Gambar 3.23 merupakan halaman *form* pemasangan papan nama yang digunakan oleh *admin* fasilitas kelengkapan untuk menambahkan atau meng-*edit* data. Jika semua *field* pada *form* sudah terisi dan menekan tombol simpan, maka data tersimpan ke dalam *database* dan jika tidak jadi menambahkan atau meng-*edit* data maka tekan tombol batal dan kembali ke halaman pada Gambar 3.22.

6. Tampilan halaman menu data jenis rambu

Admin – Sistem Informasi. x +
https://localhost/project_akhir/Rambu/data_jenisrambu
SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Papan Nama
- Data Pemasangan Papan Nama
- Data Jenis Rambu Lalu Lintas
- Data Pemasangan Rambu Lalu Lintas
- Data Pemasangan APILL
- Data Pemasangan RPPJ
- List Pengaduan
- Pengumuman
- Laporan

Data Jenis Rambu

Gambar	Kategori	Kode dan Keterangan Rambu	Aksi
			<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 3.24 Tampilan halaman menu data jenis rambu.

Gambar 3.24 merupakan halaman menu data jenis rambu yang digunakan oleh *admin* fasilitas kelengkapan. Pada halaman ini menampilkan tabel untuk jenis rambu. Ada 3 proses yang dapat dilakukan dalam halaman ini, yaitu tambah data baru, *edit* data, dan hapus data.

7. Tampilan halaman *form* jenis rambu



Form Jenis Rambu

Kode Rambu

Kategori Rambu

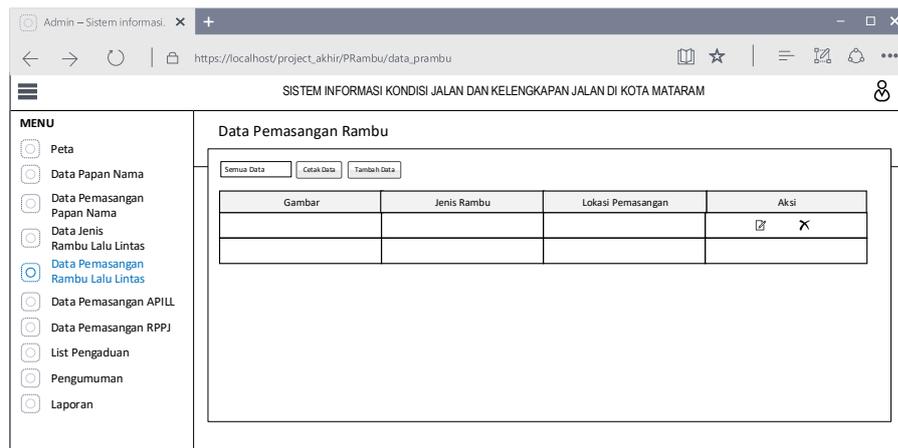
Keterangan Rambu

Gambar Rambu

Gambar 3.25 Tampilan halaman *form* jenis rambu.

Gambar 3.25 merupakan halaman *form* jenis rambu yang digunakan oleh *admin* fasilitas kelengkapan. Pada halaman ini menampilkan *form* untuk menambahkan atau meng-*edit* data.

8. Tampilan halaman menu data pemasangan rambu



Admin – Sistem informasi. x +

https://localhost/project_akhir/PRambu/data_prambu

SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Papan Nama
- Data Pemasangan Papan Nama
- Data Jenis Rambu Lalu Lintas
- Data Pemasangan Rambu Lalu Lintas**
- Data Pemasangan APILL
- Data Pemasangan RPPJ
- List Pengaduan
- Pengumuman
- Laporan

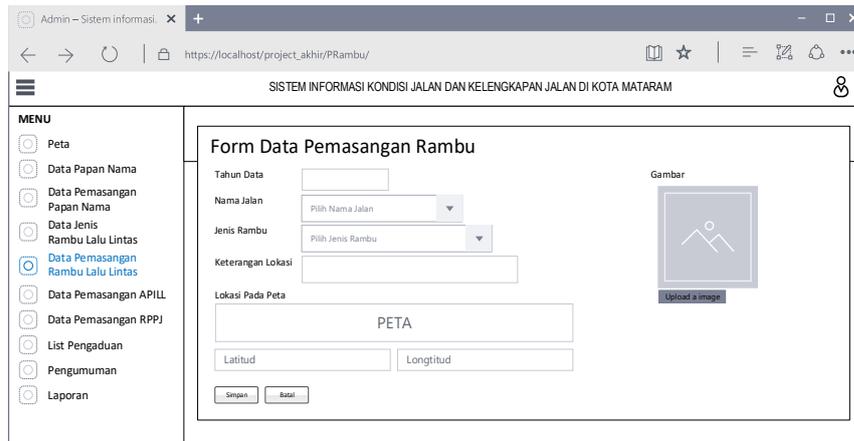
Data Pemasangan Rambu

Gambar	Jenis Rambu	Lokasi Pemasangan	Aksi
			<input type="button" value="✎"/> <input type="button" value="✕"/>

Gambar 3.26 Tampilan halaman menu data pemasangan rambu.

Gambar 3.26 merupakan halaman menu data pemasangan rambu yang digunakan oleh *admin* fasilitas kelengkapan, pada halaman ini ditampilkan semua data pemasangan yang dilakukan dalam tabel. Pada setiap data yang ada dapat dilakukan dua tindakan yaitu menghapus data, dan merubah data.

9. Tampilan halaman *form* pemasangan rambu



Gambar 3.27 Tampilan halaman *form* pemasangan rambu.

Gambar 3.27 merupakan halaman *form* pemasangan rambu yang digunakan oleh *admin* fasilitas kelengkapan untuk menambahkan atau meng-*edit* data. Jika semua *field* pada *form* sudah terisi dan menekan tombol simpan, maka data tersimpan ke dalam *database* dan jika tidak jadi menambahkan atau meng-*edit* data maka tekan tombol batal dan kembali ke halaman pada Gambar 3.26.

10. Tampilan halaman menu data pemasangan APILL

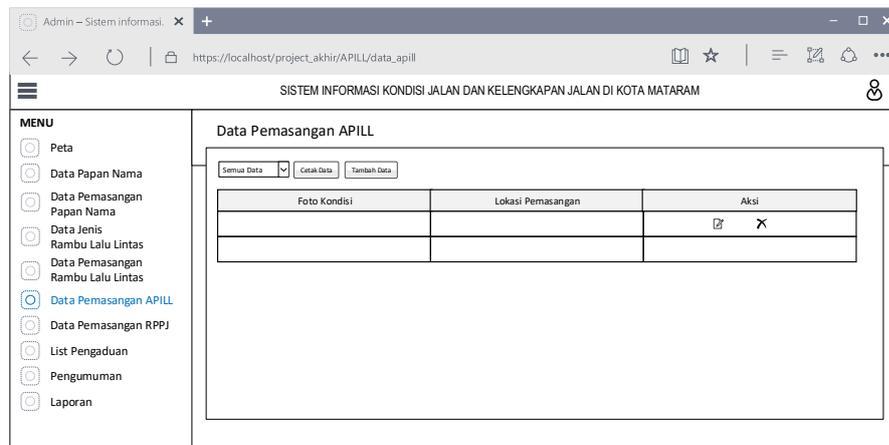


Foto Kondisi	Lokasi Pemasangan	Aksi
		 

Gambar 3.28 Tampilan halaman menu data pemasangan APILL.

Gambar 3.28 merupakan halaman menu data pemasangan APILL yang digunakan oleh *admin* fasilitas kelengkapan, pada halaman ini ditampilkan semua data pemasangan yang dilakukan dalam tabel. Pada setiap data yang ada dapat dilakukan dua tindakan yaitu menghapus data, dan merubah data.

11. Tampilan halaman form pemasangan APILL

The screenshot shows a web browser window with the URL `https://localhost/project_akhir/APILL/`. The page title is "SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM". On the left, there is a "MENU" sidebar with options: Peta, Data Papan Nama, Data Pemasangan Papan Nama, Data Jenis Rambu Lalu Lintas, Data Pemasangan Rambu Lalu Lintas, Data Pemasangan APILL (highlighted), Data Pemasangan RPPJ, List Pengaduan, Pengumuman, and Laporan. The main content area is titled "Form Data Pemasangan APILL" and contains the following fields: "Tahun Data" (text input), "Nama Jalan" (dropdown menu with "Pilih Nama Jalan"), "Keterangan Lokasi" (text area), "Lokasi Pada Peta" (map area labeled "PETA"), "Latitud" (text input), and "Longitud" (text input). There are "Simpan" and "Batal" buttons at the bottom. On the right, there is a "Gambar" section with a placeholder image and an "Upload a image" button.

Gambar 3.29 Tampilan halaman *form* pemasangan APILL.

Gambar 3.29 merupakan halaman *form* pemasangan APILL yang digunakan oleh *admin* fasilitas kelengkapan untuk menambahkan atau meng-*edit* data. Jika semua *field* pada *form* sudah terisi dan menekan tombol simpan, maka data tersimpan ke dalam *database* dan jika tidak jadi menambahkan atau meng-*edit* data maka tekan tombol batal dan kembali ke halaman pada Gambar 3.28.

12. Tampilan halaman menu data pemasangan RPPJ

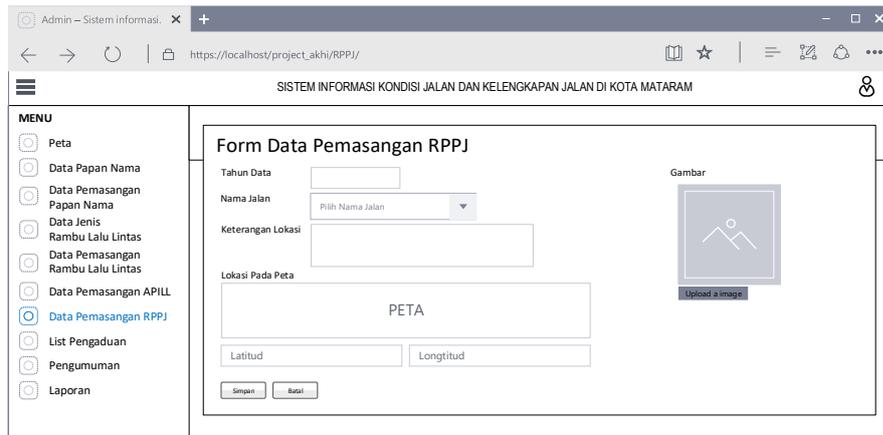
The screenshot shows a web browser window with the URL `https://localhost/project_akhir/RPPJ/data_rppj`. The page title is "SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM". On the left, there is a "MENU" sidebar with options: Peta, Data Papan Nama, Data Pemasangan Papan Nama, Data Jenis Rambu Lalu Lintas, Data Pemasangan Rambu Lalu Lintas, Data Pemasangan APILL, Data Pemasangan RPPJ (highlighted), List Pengaduan, Pengumuman, and Laporan. The main content area is titled "Data Pemasangan RPPJ" and contains a table with columns: "Foto Kondisi", "Lokasi Pemasangan", and "Aksi". The "Aksi" column has icons for edit and delete. There are also buttons for "Semua Data", "Cetak Data", and "Tambah Data" at the top of the table.

Foto Kondisi	Lokasi Pemasangan	Aksi

Gambar 3.30 Tampilan halaman menu data pemasangan RPPJ.

Gambar 3.30 merupakan halaman menu data pemasangan RPPJ yang digunakan oleh *admin* fasilitas kelengkapan, pada halaman ini ditampilkan semua data pemasangan yang dilakukan dalam tabel. Pada setiap data yang ada dapat dilakukan dua tindakan yaitu menghapus data, dan merubah data.

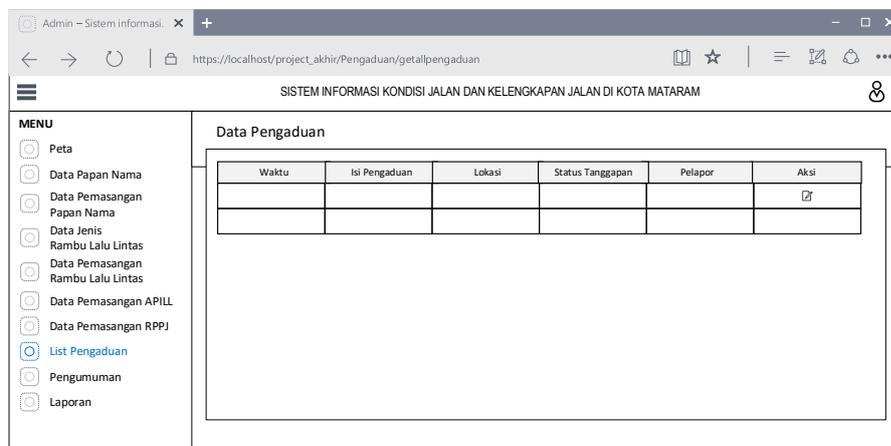
13. Tampilan halaman form pemasangan RPPJ



Gambar 3.31 Tampilan halaman *form* pemasangan RPPJ.

Gambar 3.31 merupakan halaman *form* pemasangan RPPJ yang digunakan oleh *admin* fasilitas kelengkapan untuk menambahkan atau meng-*edit* data. Jika semua *field* pada *form* sudah terisi dan menekan tombol simpan, maka data tersimpan ke dalam *database* dan jika tidak jadi menambahkan atau meng-*edit* data maka tekan tombol batal dan kembali ke halaman pada Gambar 3.30.

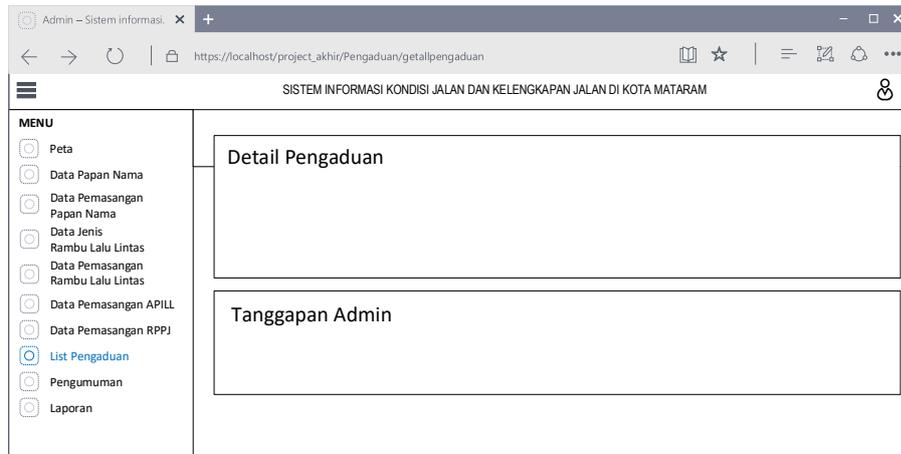
14. Tampilan halaman menu pengaduan kerusakan faskel



Gambar 3.32 Tampilan halaman menu pengaduan kerusakan faskel.

Gambar 3.32 merupakan halaman menu pengaduan yang digunakan oleh *admin* faskel, pada halaman ini ditampilkan semua pengaduan kerusakan yang dikirim oleh pengguna aplikasi *mobile* dalam bentuk tabel. Terdapat tombol aksi pada setiap pengaduan untuk mengarahkan ke halaman lain untuk melihat detail pengaduan.

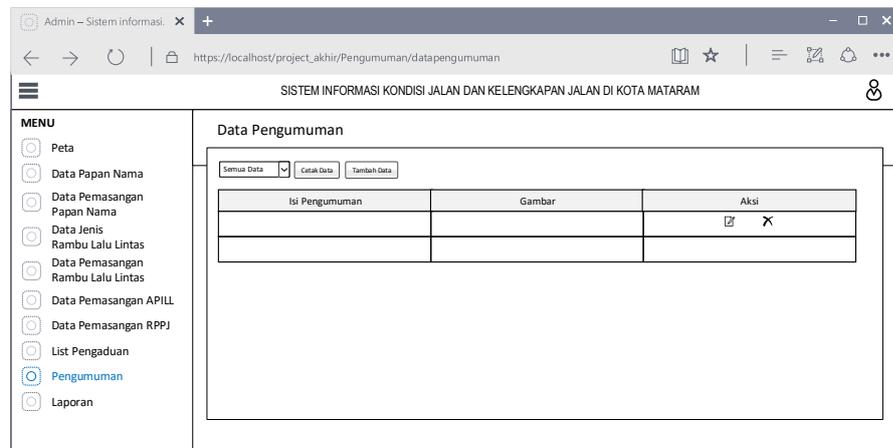
15. Tampilan halaman detail pengaduan admin faskel



Gambar 3.33 Tampilan halaman detail pengaduan admin faskel.

Gambar 3.33 merupakan halaman detail pengaduan yang digunakan oleh *admin* faskel, pada halaman ini menampilkan detail dari salah satu pengaduan yang ada. Pada halaman ini juga terdapat *form* yang digunakan untuk melakukan *feedback* dari pengaduan tersebut dan menekan tombol Kirim.

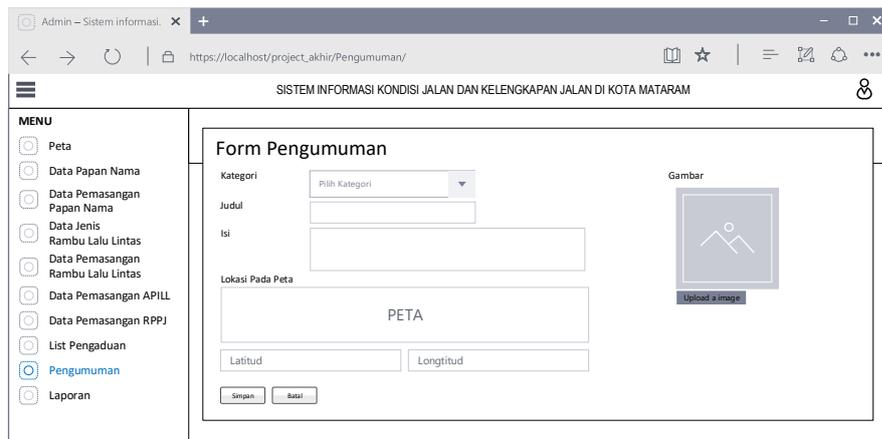
16. Tampilan halaman menu pengumuman admin faskel



Gambar 3.34 Tampilan halaman menu pengumuman admin faskel.

Gambar 3.34 merupakan halaman menu pengumuman, pada halaman ini menampilkan daftar pengumuman dalam bentuk tabel yang pernah dibuat oleh *admin*. Terdapat 3 tindakan yang dapat dilakukan pada halaman ini yaitu tombol untuk ke halaman menambahkan pengumuman baru, meng-*edit* isi data atau status pengumuman, dan menghapus pengumuman.

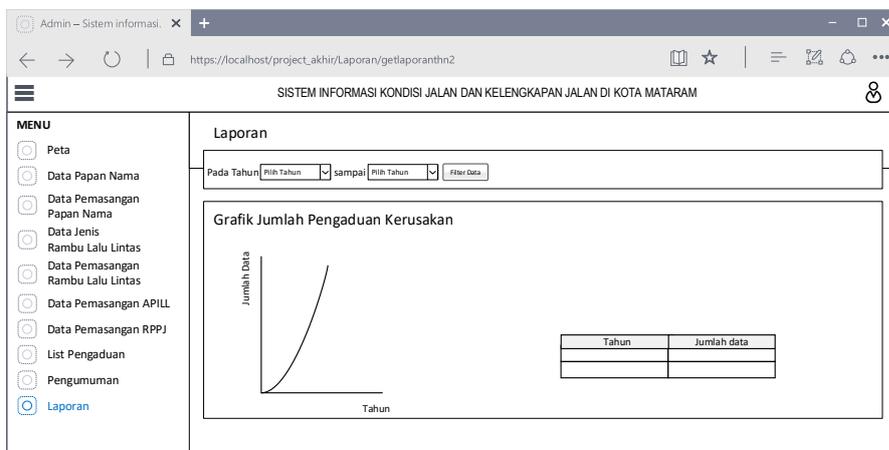
17. Tampilan halaman *form* pengumuman admin faskel



Gambar 3.35 Tampilan halaman *form* pengumuman admin faskel.

Gambar 3.35 merupakan halaman *form* pengumuman yang digunakan oleh *admin* untuk menambahkan atau meng-*edit* pengumuman. Menekan tombol simpan jika semua *field* yang sudah diisi, jika tidak jadi menambahkan atau meng-*edit* data maka menekan tombol batal dan kembali ke tampilan pada Gambar 3.34.

18. Tampilan halaman menu laporan admin faskel



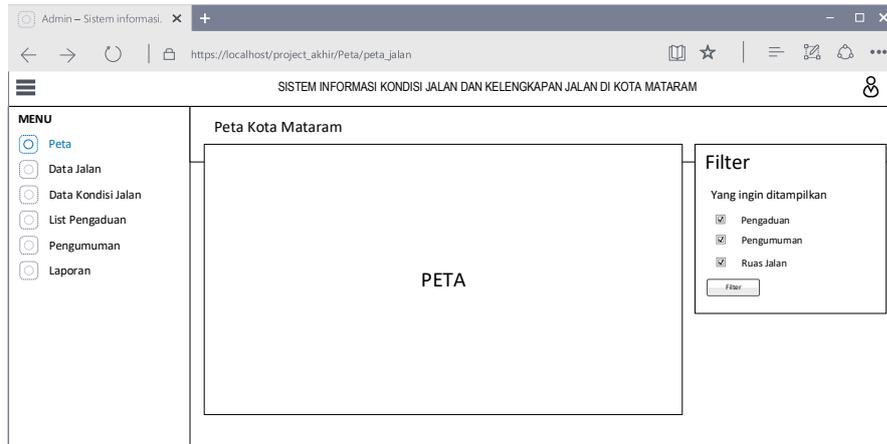
Gambar 3.36 Tampilan halaman menu laporan admin faskel.

Gambar 3.36 merupakan halaman menu laporan, pada halaman ini menampilkan laporan dari keseluruhan data dari pendataan dan juga pengaduan kerusakan digunakan oleh *admin* faskel. Data ditampilkan dalam bentuk grafik dan tabel, dimana dapat dipilih jenis laporan dan data yang ditampilkan dapat berupa data keseluruhan atau per-tahunnya. Serta data pada tabel dapat dilihat detailnya.

3.2.3.7.2 Rancangan tampilan aplikasi *web* untuk admin jalan

Perancangan tampilan aplikasi *web* yang akan digunakan oleh *admin* jalan atau Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram terdiri dari beberapa tampilan sebagai berikut:

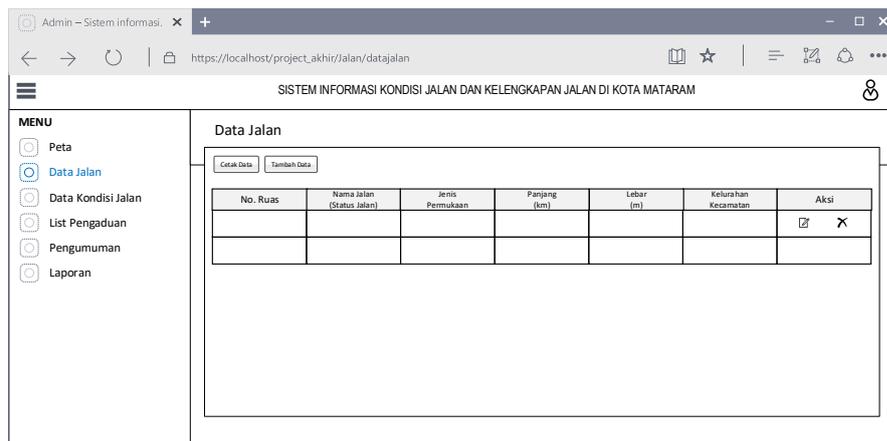
1. Tampilan halaman menu peta untuk admin jalan



Gambar 3.37 Tampilan halaman menu peta untuk admin jalan.

Gambar 3.37 merupakan menu utama untuk admin jalan setelah berhasil *login*. Menampilkan sebuah peta dan juga koordinat yang berkaitan dengan jalan jalan seperti koordinat ruas jalan, jalan yang rusak, dan jalan yang akan diperbaiki.

2. Tampilan halaman menu data jalan



Gambar 3.38 Tampilan halaman menu data jalan.

Gambar 3.38 merupakan halaman menu data jalan yang digunakan oleh *admin* jalan. Pada halaman ini menampilkan tabel untuk data jalan. Ada 3 proses yang dapat dilakukan dalam halaman ini, yaitu tambah data baru, *edit* data, dan hapus data.

3. Tampilan halaman *form* data jalan

Admin - Sistem informasi. x +
https://localhost/project_akhir/jalan/
SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Jalan
- Data Kondisi Jalan
- List Pengaduan
- Pengumuman
- Laporan

Form Data Jalan

Peta

PETA

Kode Jalan

Nama Ruas

Panjang Lebar

Kecamatan

Kelurahan

Jenis Permukaan

Status Jalan

Koordinat

Simpan Batal

Gambar 3.39 Tampilan halaman *form* data jalan.

Gambar 3.39 merupakan halaman *form* data jalan yang digunakan oleh *admin* jalan. Pada halaman ini menampilkan *form* untuk menambahkan data atau meng-*edit* data. Jika tidak jadi untuk menambahkan atau meng-*edit* data maka menekan tombol batal sehingga akan keluar dari tampilan *form* ini dan kembali ke tampilan pada Gambar 3.38.

4. Tampilan halaman data kondisi jalan

Admin - Sistem informasi. x +
https://localhost/project_akhir/Kondisi_jalan/kondisijalan
SISTEM INFORMASI KONDISI JALAN DAN KELENGKAPAN JALAN DI KOTA MATARAM

MENU

- Peta
- Data Jalan
- Data Kondisi Jalan
- List Pengaduan
- Pengumuman
- Laporan

Data Kondisi Jalan

Semua Data Cetak Data Tambah Data

Nama Jalan	Baik	Sedang	Rusak Ringan	Rusak Berat	Tahun Data	Aksi

Gambar 3.40 Tampilan halaman data kondisi jalan.

Gambar 3.40 merupakan halaman data kondisi jalan, pada halaman ini menampilkan data kondisi jalan yang digunakan oleh *admin* jalan. Pada halaman ini setiap data yang ada dapat dilakukan dua tindakan yaitu menghapus data dan merubah data. Dan terdapat tombol untuk menambahkan data.

5. Tampilan halaman *form* tambah data kondisi jalan

Nama Jalan	Baik	Sedang	Rusak Ringan	Rusak Berat

Gambar 3.41 Tampilan halaman form tambah data kondisi jalan.

Gambar 3.41 merupakan halaman *form* data kondisi jalan yang digunakan oleh *admin* jalan untuk menambahkan data kondisi jalan. Jika semua *field* sudah terisi dan menekan tombol simpan dan jika tidak jadi menambahkan data maka tekan tombol batal dan kembali ke tampilan pada Gambar 3.40.

6. Tampilan halaman *form* edit data kondisi jalan

Edit Data Kondisi Jalan

Nama Jalan

Panjang X Lebar X

Kondisi Jalan

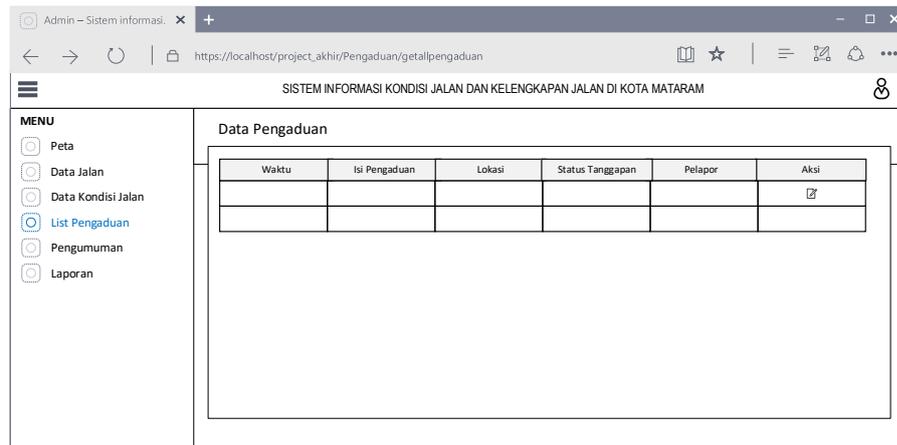
Baik Sedang Rusak Ringan Rusak Berat

Simpan Batal

Gambar 3.42 Tampilan halaman form edit data kondisi jalan.

Gambar 3.42 merupakan halaman *form* data kondisi jalan yang digunakan oleh *admin* jalan untuk meng-*edit* data kondisi jalan. Jika semua *field* sudah terisi dan menekan tombol simpan dan jika tidak jadi meng-*edit* data maka tekan tombol batal dan kembali ke tampilan pada Gambar 3.40.

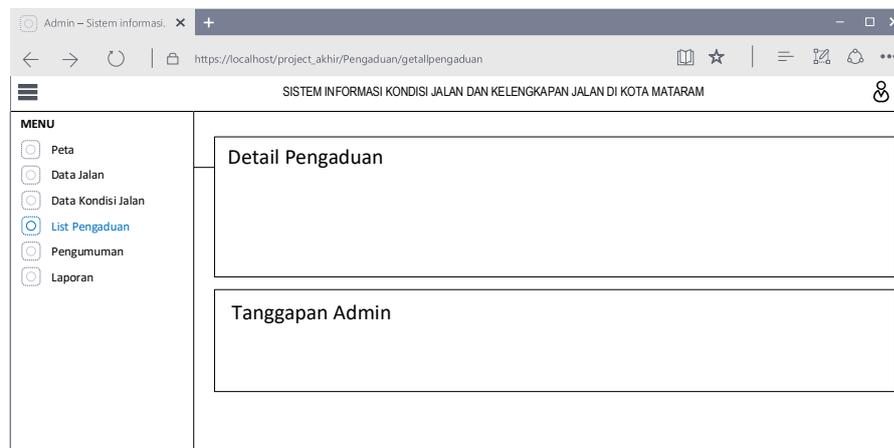
7. Tampilan halaman menu pengaduan kerusakan jalan



Gambar 3.43 Tampilan halaman menu pengaduan kerusakan jalan.

Gambar 3.43 merupakan halaman menu pengaduan yang digunakan oleh *admin* jalan, pada halaman ini ditampilkan semua pengaduan kerusakan yang dikirim oleh pengguna aplikasi *mobile* dalam bentuk tabel. Terdapat tombol aksi pada setiap pengaduan untuk mengarahkan ke halaman lain untuk melihat detail pengaduan.

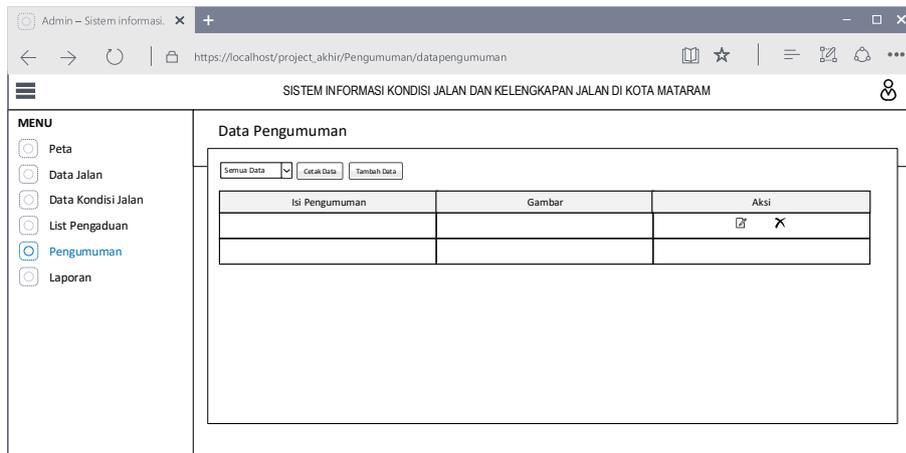
8. Tampilan halaman detail pengaduan admin jalan



Gambar 3.44 Tampilan halaman detail pengaduan admin jalan.

Gambar 3.44 merupakan halaman detail pengaduan yang digunakan oleh *admin* jalan, pada halaman ini menampilkan detail dari salah satu pengaduan yang ada. Pada halaman ini juga terdapat *form* yang digunakan untuk melakukan *feedback* dari pengaduan tersebut dan menekan tombol Kirim.

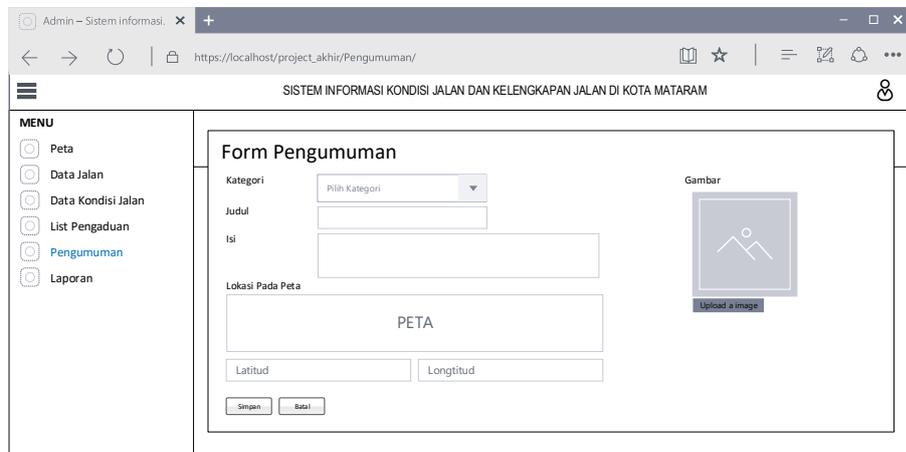
9. Tampilan halaman menu pengumuman admin jalan



Gambar 3.45 Tampilan halaman menu pengumuman admin jalan.

Gambar 3.45 merupakan halaman menu pengumuman, pada halaman ini menampilkan daftar pengumuman dalam bentuk tabel yang pernah dibuat oleh *admin*. Terdapat 3 tindakan yang dapat dilakukan pada halaman ini yaitu tombol untuk ke halaman menambahkan pengumuman baru, meng-*edit* isi data atau status pengumuman, dan menghapus pengumuman.

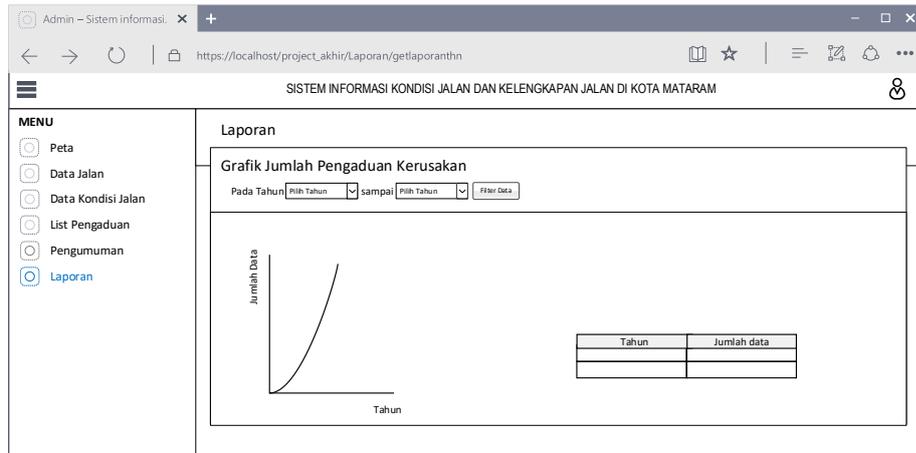
10. Tampilan halaman *form* pengumuman admin jalan



Gambar 3.46 Tampilan halaman *form* pengumuman admin jalan.

Gambar 3.46 merupakan halaman *form* pengumuman yang digunakan oleh *admin* untuk menambahkan atau meng-*edit* pengumuman. Menekan tombol simpan jika semua *field* yang sudah diisi, jika tidak jadi menambahkan atau meng-*edit* data maka menekan tombol batal dan kembali ke tampilan pada Gambar 3.50.

11. Tampilan halaman menu laporan admin jalan



Gambar 3.47 Tampilan halaman menu laporan admin jalan.

Gambar 3.47 merupakan halaman menu laporan, pada halaman ini menampilkan laporan dari keseluruhan data dari pendataan dan juga pengaduan kerusakan digunakan oleh *admin* jalan. Data ditampilkan dalam bentuk grafik dan tabel, dimana dapat dipilih jenis laporan dan data yang ditampilkan dapat berupa data keseluruhan atau per-tahunnya. Serta data pada tabel dapat dilihat detailnya.

3.2.3.7.3 Rancangan tampilan aplikasi *mobile* untuk masyarakat

Perancangan tampilan aplikasi *mobile* yang akan digunakan oleh masyarakat Kota Mataram terdiri dari beberapa tampilan sebagai berikut:

1. Tampilan halaman *login*



Gambar 3.48 Tampilan halaman *login* aplikasi mobile.

Gambar 3.48 merupakan halaman *login* aplikasi yang digunakan oleh masyarakat Kota Mataram untuk dapat masuk ke dalam aplikasi dengan memasukkan *username* dan *password*, selanjutnya memilih tombol *login*. Jika *username* dan *password*

terdaftar dalam *database* maka akan menampilkan halaman utama tapi jika tidak terdaftar maka akan menampilkan pesan *error*. Jika belum memiliki akun maka dapat mendaftar dengan mengklik tombol *Sign Up*.

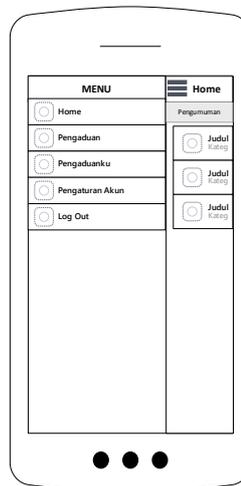
2. Tampilan halaman *sign up*



Gambar 3.49 Tampilan halaman *sign up*.

Gambar 3.49 merupakan halaman *sign up* yang digunakan oleh masyarakat Kota Mataram untuk mendaftarkan akun ke dalam aplikasi. Pada halaman ini menampilkan *form* yang harus diisi semua *field*-nya dan menekan tombol Simpan.

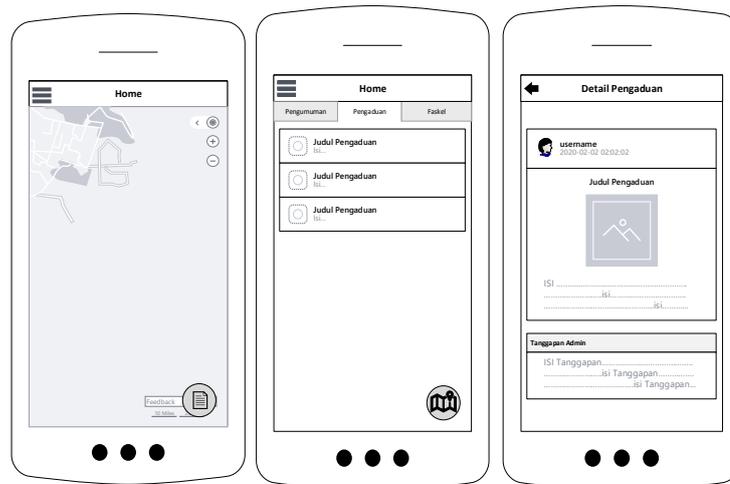
3. Tampilan navigasi aplikasi



Gambar 3.50 Tampilan navigasi aplikasi.

Gambar 3.50 merupakan tampilan navigasi setelah *login* yang digunakan oleh masyarakat. Terdapat beberapa menu yaitu menu *home*, menu pengaduan, menu pengaduanku, menu pengaturan akun, dan menu *logout*.

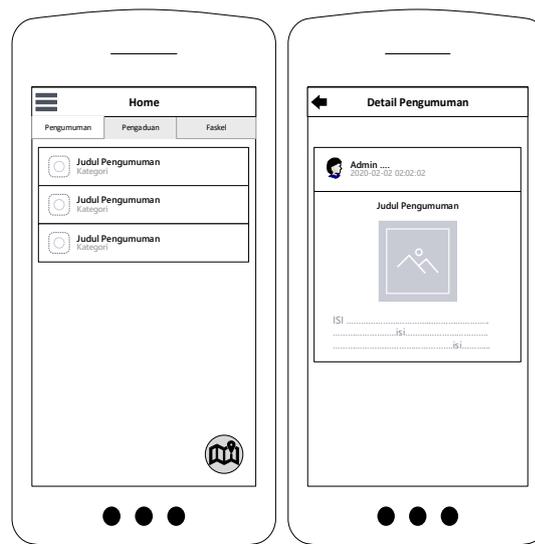
4. Tampilan halaman menu *home*



Gambar 3.51 Tampilan halaman menu *home* dan view detail pengaduan.

Gambar 3.51 merupakan halaman menu Home aplikasi yang digunakan oleh masyarakat Kota Mataram untuk melihat kondisi Kota Mataram yang ditampilkan dalam bentuk peta beserta titik-titik posisi (kiri) pengaduan kerusakan, posisi fasilitas kelengkapan yang terpasang dan posisi jika ada perbaikan atau kegiatan yang berkaitan dengan dinas yang menangani jalan dan fasilitas kelengkapan jalan. Dan juga menampilkan dalam bentuk *list* pengaduan beserta tab untuk kehalaman *list* pengumuman dan list fasilitas kelengkapan jalan (tengah) yang mengarah ke halaman *view* detail pengaduan (kanan).

5. Tampilan halaman tab pengumuman *admin*

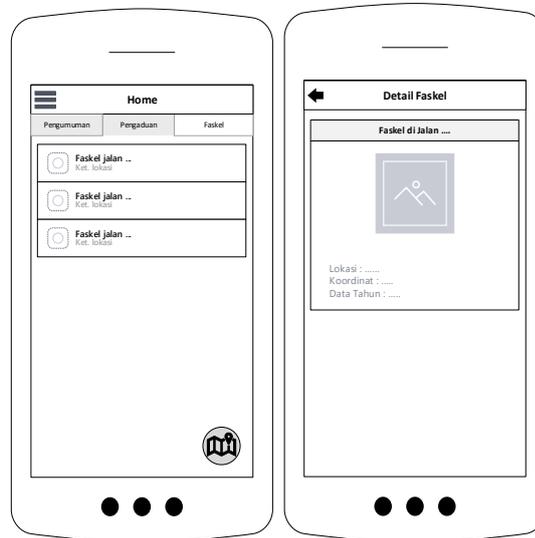


Gambar 3.52 Tampilan halaman menu tab pengumuman.

Gambar 3.52 merupakan tampilan halaman menu tab pengumuman yang digunakan untuk melihat *list* pengumuman (kiri) perencanaan untuk perbaikan dan lain-lainnya

yang akan dilakukan dan mengharuskan masyarakat terutama pengendara angkutan darat untuk berhati-hati, dan pengumuman tersebut dapat di lihat detail isi pengumumannya (kanan).

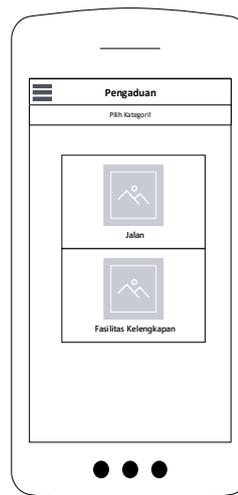
6. Tampilan halaman info kondisi fasilitas kelengkapan



Gambar 3.53 Tampilan tab menu faskel jalan.

Gambar 3.53 merupakan tampilan halaman tab menu faskel jalan yang digunakan untuk melihat *list* kondisi fasilitas kelengkapan (kiri) yang telah diinputkan oleh *admin* yang menangani fasilitas kelengkapan dan info tersebut dapat di lihat detail isinya (kanan).

7. Tampilan menu Laporan



Gambar 3.54 Tampilan halaman menu pengaduan.

Gambar 3.54 merupakan tampilan dari konten menu pengaduan yang digunakan oleh masyarakat. Pada konten tersebut menampilkan kategori dari pengaduan yang akan dilakukan yaitu kategori jalan dan juga kategori perlengkapan jalan.

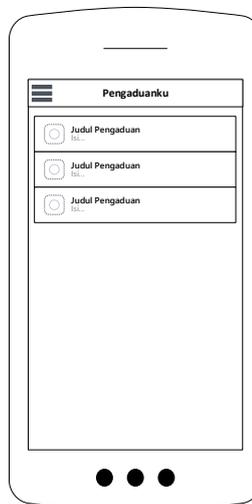
8. Tampilan *form* pengaduan kerusakan

A wireframe illustration of a mobile application screen for a complaint form. At the top left, there is a back arrow icon. The main content area is titled "Form Pengaduan" and contains a text input field for "Judul Pengaduan", a larger text area for "Isi Pengaduan", a button labeled "Upload Foto", and a prominent "KIRIM" button at the bottom. The screen is framed by a rounded rectangle with three dots at the bottom center, representing a mobile device.

Gambar 3.55 Tampilan *form* pengaduan kerusakan.

Gambar 3.55 merupakan tampilan *form* pengaduan kerusakan Fasilitas Kelengkapan ataupun kerusakan Jalan yang digunakan oleh masyarakat Kota Mataram. Pada konten tersebut menampilkan *form* yang harus diisi semua *field*-nya kemudian dikirim ke *admin* Fasilitas Kelengkapan ataupun *admin* Jalan.

9. Tampilan menu pengaduanku



Gambar 3.56 Tampilan halaman menu pengaduanku.

Gambar 3.56 merupakan tampilan dari konten menu pengaduanku yang digunakan oleh masyarakat Kota Mataram. Pada konten tersebut menampilkan semua kategori laporan yang telah dilakukan oleh *user* tersebut.

10. Tampilan menu pengaturan akun



Gambar 3.57 Tampilan halaman menu pengaturan akun.

Gambar 3.57 merupakan tampilan dari konten menu pengaturan akun yang digunakan oleh masyarakat Kota Mataram. Pada konten tersebut menampilkan data dari akun yang diinputkan masyarakat saat melakukan *signup*. Data dapat diganti kemudian menekan tombol *update data*.

3.2.4 Implementasi *Coding* aplikasi

Pada tahapan ini dilakukan proses implementasi *coding* dari rancangan sistem yang telah dilakukan ke dalam bentuk bahasa pemrograman. Sistem yang digunakan oleh *admin* akan dibangun menggunakan *framework* CodeIgniter. Sedangkan aplikasi *mobile* yang digunakan oleh masyarakat Kota Mataram dibangun menggunakan *framework* Ionic.

3.2.5 Pengujian aplikasi

Teknik pengujian aplikasi yang akan digunakan pada penelitian ini terdiri dari pengujian *black box* dan pengujian MOS (*Mean Opinion Score*).

Pengujian *black box* merupakan pengujian untuk mengetahui apakah semua fungsi aplikasi telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan yaitu aplikasi kondisi jalan dan fasilitas kelengkapan jalan, yang digunakan masyarakat melihat kondisi jalan dan fasilitas kelengkapan jalan, serta mengirim pengaduan kerusakan ke dinas yang bertanggung jawab dalam bentuk aplikasi *mobile* dan juga aplikasi *web* yang digunakan oleh dinas untuk menerima pengaduan tersebut dan memberi informasi perbaikan yang akan dilakukan. Aspek yang diuji dan detail pengujian yang dilakukan dengan metode *black box* pada aplikasi ini tersaji pada Tabel 3.15.

Tabel 3.15 Rencana pengujian *black box*.

No	Item	Detail Uji
1.	<i>Login</i>	Masuk ke sistem dengan memasukkan <i>username</i> dan <i>password</i> yang benar
		Masuk ke sistem dengan memasukkan <i>username</i> dan <i>password</i> yang salah
2.	Mengelola data jalan	Menambahkan data jalan
		Meng- <i>edit</i> data jalan
		Menghapus data jalan
3.	Mengelola kondisi jalan	Menambahkan kondisi jalan
		Meng- <i>edit</i> kondisi jalan
		Menghapus kondisi jalan
4.	Mengelola data papan nama jalan	Menambahkan data papan nama jalan
		Meng- <i>edit</i> data papan nama jalan
		Menghapus data papan nama jalan
5.	Mengelola data pemasangan papan nama	Menambahkan data pemasangan
		Meng- <i>edit</i> data pemasangan
		Menghapus data pemasangan
6.	Mengelola data jenis rambu lalu lintas	Menambahkan jenis rambu
		Meng- <i>edit</i> jenis rambu
		Menghapus jenis rambu
7.	Mengelola data pemasangan rambu	Menambahkan data pemasangan
		Meng- <i>edit</i> data pemasangan
		Menghapus data pemasangan
8.	Mengelola data pemasangan APILL	Menambahkan data pemasangan
		Meng- <i>edit</i> data pemasangan
		Menghapus data pemasangan
9.	Mengelola data pemasangan RPPJ	Menambahkan data pemasangan
		Meng- <i>edit</i> data pemasangan
		Menghapus data pemasangan
10.	Mengelola pengumuman	Menambahkan pengumuman
		Meng- <i>edit</i> pengumuman
		Menghapus pengumuman
11.	Mengelola pengaduan	Melihat pengaduan-pengaduan yang dilakukan masyarakat dan melakukan tanggapan terhadap pengaduan yang telah dilakukan.
12.	Melihat laporan tahunan oleh <i>admin</i> jalan	Menampilkan laporan tahunan pengaduan kerusakan jalan dan kondisi jalan berupa grafik dan tabel.
13.	Melihat laporan tahunan oleh <i>admin</i> faskel	Menampilkan laporan tahunan pengaduan kerusakan fasilitas kelengkapan jalan dan pemasangan fasilitas kelengkapan jalan berupa grafik dan tabel.
14.	Melihat kondisi jalan dan kelengkapan jalan	Melihat fasilitas kelengkapan jalan yang terpasang.
		Melihat pengumuman dari dinas.
		Melihat pengaduan kerusakan yang dilakukan masyarakat.
15.	Melakukan pengaduan kerusakan.	Proses melakukan pengaduan kerusakan oleh masyarakat.

Mean Opinion Score (MOS) merupakan penilaian subjektif dengan menggunakan kuisisioner yang melibatkan responden dalam sistem. Kuisisioner dilakukan untuk

mengetahui karakteristik dan performa dari sistem yang dibuat serta kemudahan dalam penggunaan sistem. Pengujian menggunakan MOS ini dilakukan oleh pihak yang bersangkutan pada lembaga pemerintah yang bertanggungjawab menangani masalah jalan ataupun fasilitas kelengkapan jalan yaitu dinas Perhubungan dan Bina Marga di Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram yang akan menggunakan aplikasi *web* untuk memberikan informasi perbaikan, informasi kondisi jalan dan kelengkapan jalan, serta menerima pengaduan kerusakan dari masyarakat. Masyarakat yang akan menggunakan aplikasi *mobile* untuk mengirim pengaduan kerusakan dan juga untuk mempermudah masyarakat mendapat informasi tentang kondisi jalan dan kondisi kelengkapan jalan yang ada di Kota Mataram. Kuesioner disebarakan kepada 3 orang perwakilan dari Dinas Perhubungan, 3 orang Dinas Pekerjaan Umum dan Penataan Ruang dan 24 orang dari masyarakat umum. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan aplikasi. Dan dengan menggunakan standar penilaian MOS terhadap pengujian aplikasi dapat diketahui apakah aplikasi sudah sesuai dengan tujuan dan manfaat yang diharapkan.

3.2.6 Dokumentasi

Sistem yang telah berfungsi dengan baik dan sesuai dengan rancangan yang ada, akan didokumentasikan dalam bentuk laporan. Laporan tersebut terdiri dari pendahuluan, tinjauan pustaka dan dasar teori, metode perancangan, hasil dan pembahasan, serta kesimpulan dan saran.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Pada bab ini, akan dibahas hasil pembangunan “Sistem Informasi Hybrid Kondisi Jalan dan Fasilitas Kelengkapan Jalan di Kota Mataram”. Pada pembangunan yang dilakukan telah sesuai dengan perencanaan yang dijabarkan pada bab sebelumnya. Pembahasan yang akan dijelaskan meliputi implementasi *database*, implementasi *class*, dan implementasi *interface* yang dibuat.

4.1.1 Implementasi Database

Implementasi *database* dilakukan dengan terlebih dahulu membuat *database* baru untuk sistem bernama “sik_japer”. Pada *database* tersebut memiliki 14 tabel untuk menyimpan seluruh data sistem. Adapun penjelasan lengkap setiap tabelnya dapat dilihat pada lampiran II. Berikut ini merupakan daftar tabel yang ada pada *database* sistem dapat dilihat pada Gambar 4.1.

Table	Action	Rows	Type	Collation	Size	Overhead
admin	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
feedback	★ Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	-
jalan	★ Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16 KiB	-
jenis_rambu	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
kondisi_jalan	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 KiB	-
melakukan_pengaduan	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
mobile_user	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
papan_nama	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
pasang_apill	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
pasang_pn	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	-
pasang_rambu	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
pasang_rppj	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
pengaduan	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
pengumuman	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	-
14 table(s)	Sum	35	InnoDB	latin1_swedish_ci	224 KiB	0 B

Gambar 4.1 Daftar tabel pada *database* sistem.

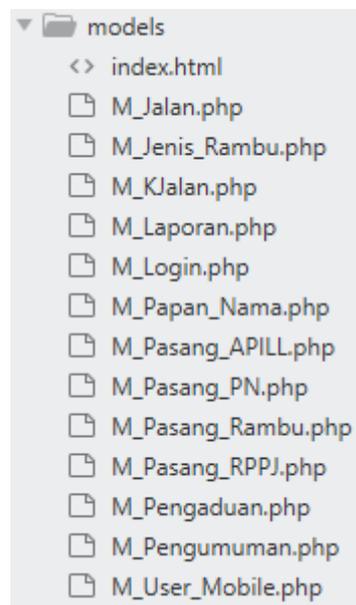
4.1.2 Implementasi Class

Implementasi *class* pada sistem ini dibagi menjadi dua bagian yaitu implementasi *class* untuk aplikasi *website* dan implementasi *class* untuk aplikasi *mobile*. Dalam perancangan *website*, *class* dibangun berdasarkan struktur pada *framework* CodeIgniter yang terdiri dari MVC (*Model, View, Controller*). Sedangkan dalam perancangan *mobile*,

class dibangun berdasarkan struktur pada *framework* ionic. Berikut merupakan implementasi *class* yang dilakukan dalam pembangunan sistem.

4.1.2.1 Implementasi *class model* pada sistem

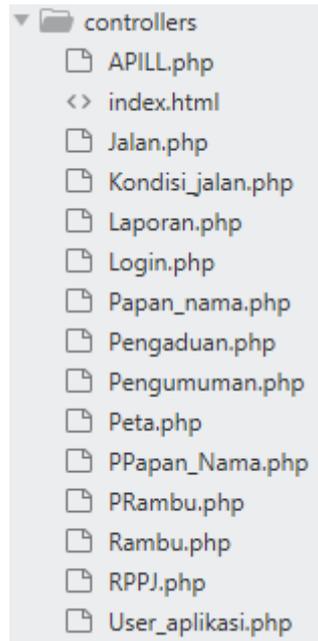
Class model digunakan untuk menghubungkan sistem dengan *database*. Keseluruhan fungsi yang terdapat di dalam *class model* merupakan berbagai *query* yang digunakan oleh sistem untuk memanipulasi data yang dibutuhkan. Pada sistem ini terdapat 13 *class model* yang dapat digunakan secara bersamaan untuk aplikasi *website* dan aplikasi *mobile*. Berikut adalah *class-class model* pada sistem yang dapat dilihat pada Gambar 4.2.



Gambar 4.2 Daftar *class model* pada sistem.

4.1.2.2 Implementasi *class controller* pada *project website*

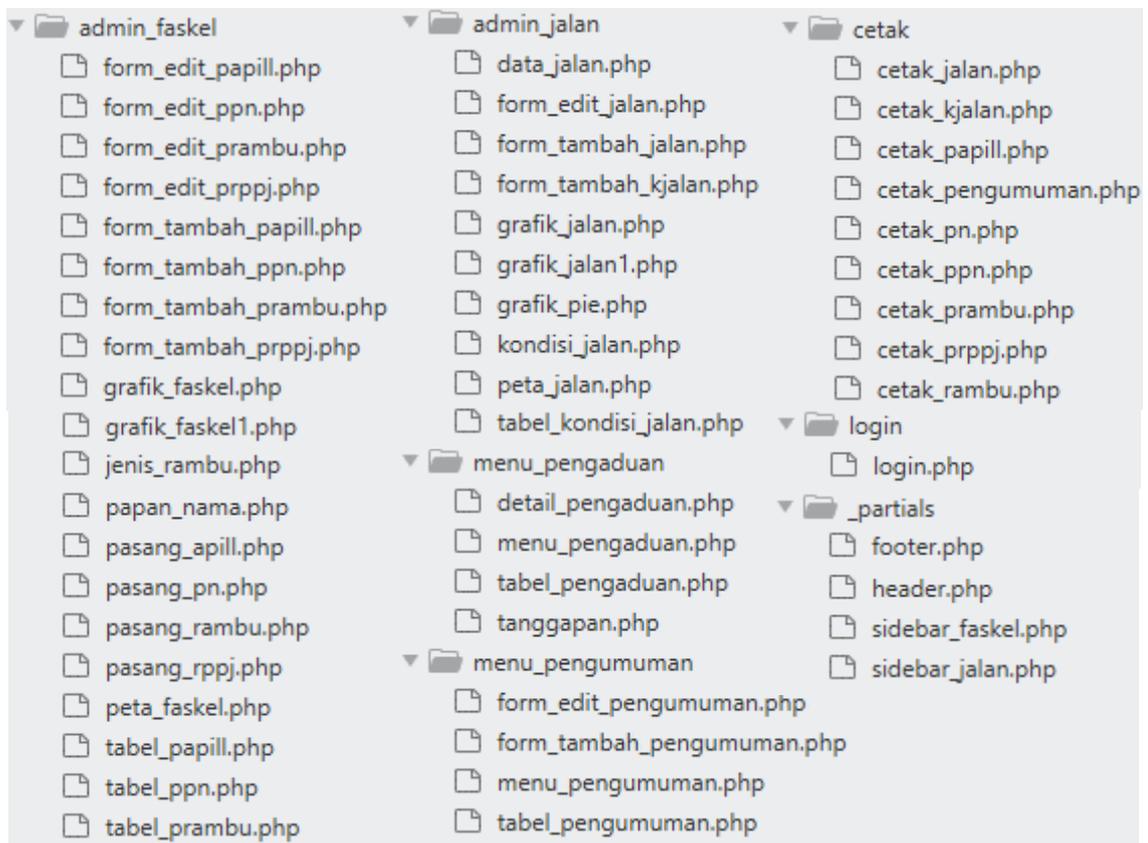
Class controller pada *project website* ini digunakan sebagai penghubung antara *model* dan *view*. *Query-query* yang berjalan pada *class model* akan ditampung pada suatu *variable* oleh *class controller* untuk ditampilkan pada *class view*. Implementasi *class controller* pada sistem ini terbagi menjadi dua bagian yaitu *class controller* yang mengembaikan data PHP dan juga *class controller* yang mengembalikan data Json. Berikut merupakan *class-class controller* pada *project website* yang dapat dilihat pada Gambar 4.3.



Gambar 4.3 Daftar *class controller* pada *project website*.

4.1.2.3 Implementasi *class view* pada *project website*

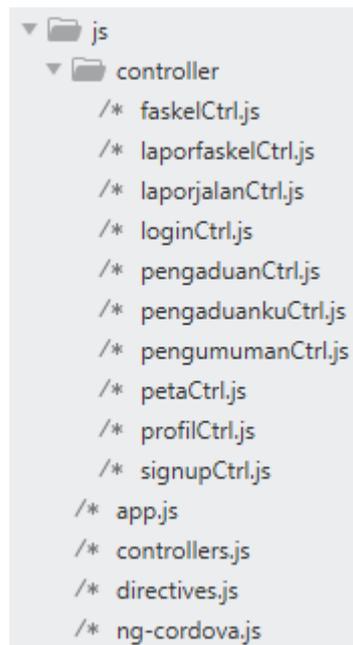
Class view pada *project website* ini digunakan untuk memberikan tampilan aplikasi *website* untuk berinteraksi dengan *end user*. Berikut adalah *class-class view* pada *project website* yang dapat dilihat pada Gambar 4.4.



Gambar 4.4 Daftar *class view* pada *project website*.

4.1.2.4 Implementasi class controller pada project mobile

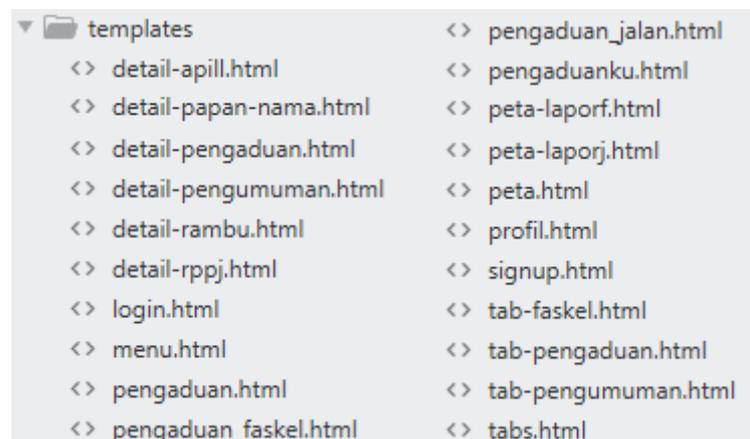
Class controller pada *project mobile* sebagai perantara untuk berinteraksi dengan sistem *server*. Dalam *class controller project mobile* terdapat kumpulan fungsi yang melakukan *request* data ke *server* dengan memanggil fungsi yang terdapat pada *class controller project website*, data tersebut disimpan dalam variabel yang kemudian ditampilkan pada *view* aplikasi *mobile*. Berikut merupakan *class-class controller* pada *project mobile* yang dapat dilihat pada Gambar 4.5.



Gambar 4.5 Daftar *class controller* pada *project mobile*.

4.1.2.5 Implementasi class view pada project mobile

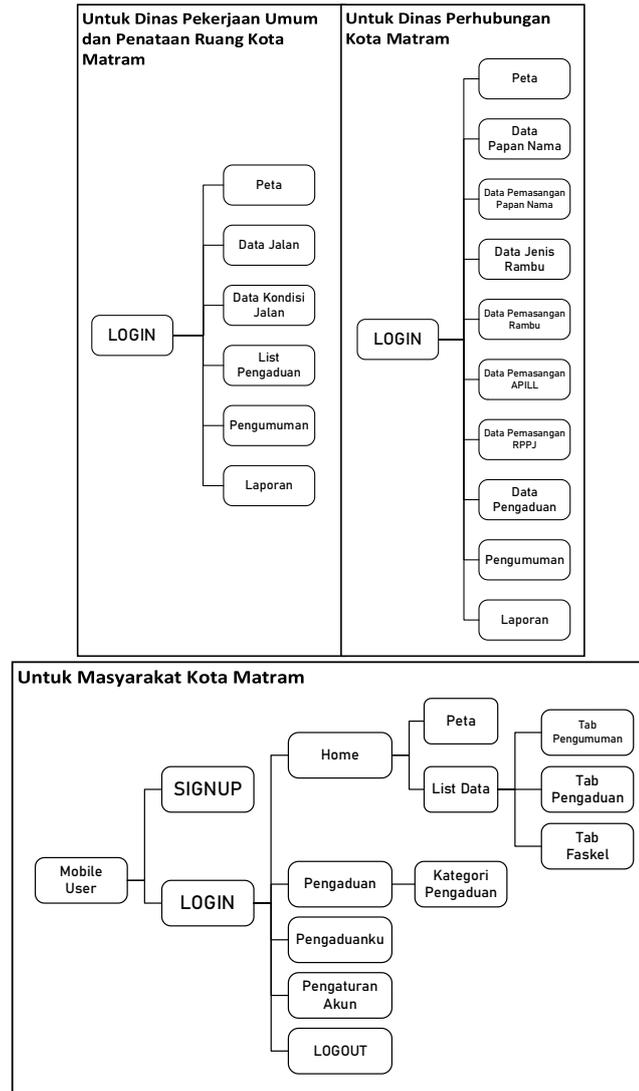
Class view pada *project mobile* ini digunakan untuk memberikan tampilan aplikasi *mobile* yang berinteraksi dengan *end user*. Berikut adalah *class-class view* pada *project mobile* dalam *folder templates* yang dapat dilihat pada Gambar 4.6.



Gambar 4.6 Daftar *class view* pada *project mobile*.

4.1.3 Implementasi Interface

Interface merupakan antar muka yang memungkinkan pengguna dapat berinteraksi dengan aplikasi. Pada sistem ini *interface* dibagi menjadi 3 yaitu *interface* aplikasi *website* untuk admin jalan, *interface* aplikasi *website* untuk admin fasilitas kelengkapan jalan, dan *interface* aplikasi *mobile* untuk masyarakat. Berikut adalah *site map* sistem yang dapat dilihat pada Gambar



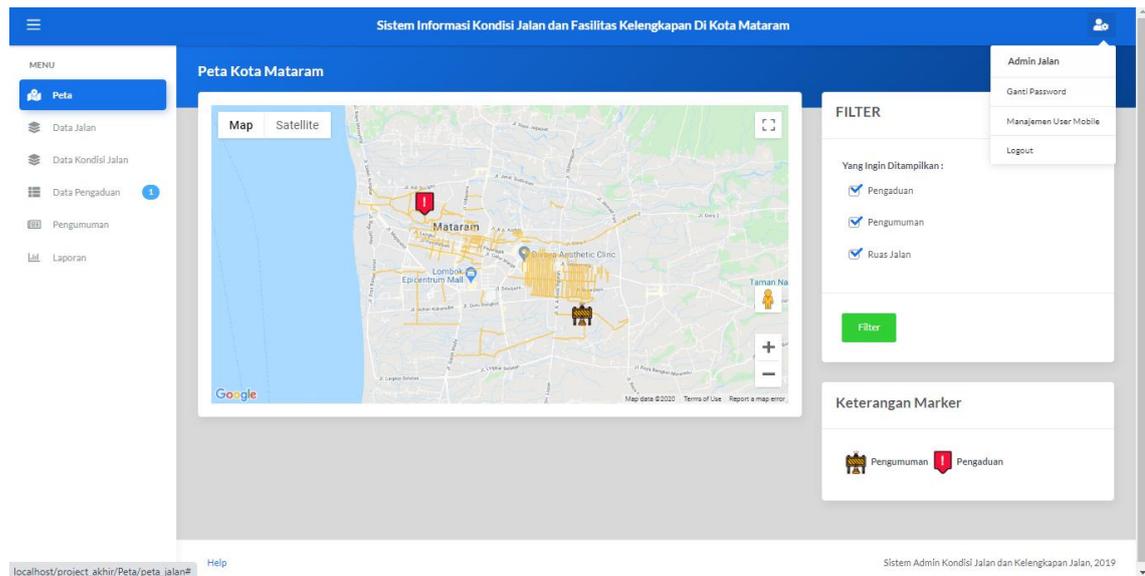
Gambar 4.7 *Site map* sistem.

4.1.3.1 *Interface* aplikasi *website* untuk admin jalan

Aplikasi *website* untuk admin jalan atau admin dari Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram terdiri dari beberapa *interface* yang diantaranya adalah sebagai berikut:

1. *Interface* menu peta untuk admin jalan

Setelah admin jalan berhasil *login*, maka menu yang pertama kali akan ditampilkan adalah menu Peta. Pada menu ini menampilkan peta yang berpusat di Kota Mataram beserta *polyline* dari ruas-ruas jalan yang sudah diinputkan oleh admin, marker yang merupakan titik koordinat lokasi pengaduan kerusakan jalan, dan marker yang merupakan titik koordinat lokasi perbaikan/pembangunan jalan. *Interface* menu peta untuk admin jalan dapat dilihat pada Gambar 4.8.



Gambar 4.8 *Interface* menu peta untuk admin jalan.

2. *Interface* menu data jalan

Menu data jalan merupakan menu yang hanya dimiliki oleh admin jalan. Pada menu ini menampilkan daftar data jalan dalam bentuk tabel yang telah di-input oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau edit data dan juga menghapus data jalan. *Interface* menu data jalan dapat dilihat pada Gambar 4.9.

No. Ruas	Nama Jalan (Status Jalan)	Jenis Permukaan	Panjang (KM)	Lebar (M)	Kelurahan Kecamatan	Action
001	Jalan Koperasi (Jalan Kota)	Hotmik	1.45	7	Ampenan Tengah Ampenan	[Edit] [Delete]
002	Jalan Lahu Mesir (Jalan Kota)	Hotmik	2.57	7	Cakranegara Selatan Baru Cakranegara	[Edit] [Delete]
003	Jalan Gora (Jalan Kota)	Hotmik	0	0	Cakranegara Utara Cakranegara	[Edit] [Delete]
004	Jalan Gora 2 (Jalan Kota)	Hotmik	0	0	Cakranegara Utara Cakranegara	[Edit] [Delete]
005	Jalan Pabean (Jalan Kota)	Hotmik	0.4	11	Dayen Pekan Ampenan	[Edit] [Delete]
006	Jalan Raden Mas Panji Anom (Jalan Kota)	Hotmik	1.55	7	Alban Tubuh Sandubaya	[Edit] [Delete]
007	Jalan Banda Seraya (Jalan Kota)	Hotmik	1.88	3.5	Pagutan Barat Mataram	[Edit] [Delete]
008	Jalan Antareja (Jalan Kota)	Hotmik	0.251	3.3	Cilinya Cakranegara	[Edit] [Delete]
009	Jalan Arjuna (Jalan Kota)	Hotmik	0.254	3	Cilinya Cakranegara	[Edit] [Delete]

Gambar 4.9 Interface menu data jalan.

3. Interface form data jalan

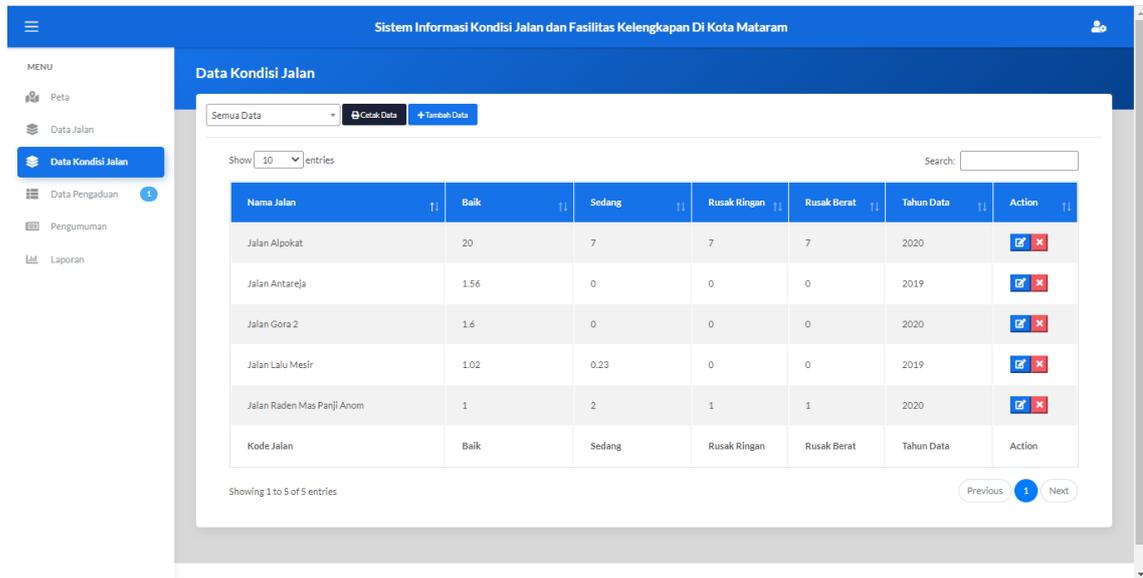
Interface form untuk tambah data jalan dan edit data jalan tidak jauh berbeda. Perbedaannya terdapat pada form edit data jalan yang memiliki field untuk koordinat polyline baru jika ingin memperbaiki gambar polyline dari jalan yang dipilih. Interface form data jalan dapat dilihat pada Gambar 4.10.

Gambar 4.10 Interface menu data jalan.

4. Interface menu data kondisi jalan

Menu data jalan kondisi merupakan menu yang hanya dimiliki oleh admin jalan. Pada menu ini menampilkan daftar data kondisi jalan dalam bentuk tabel yang telah diinput oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau

edit data dan juga menghapus data kondisi jalan. *Interface* menu data kondisi jalan dapat dilihat pada Gambar 4.11.



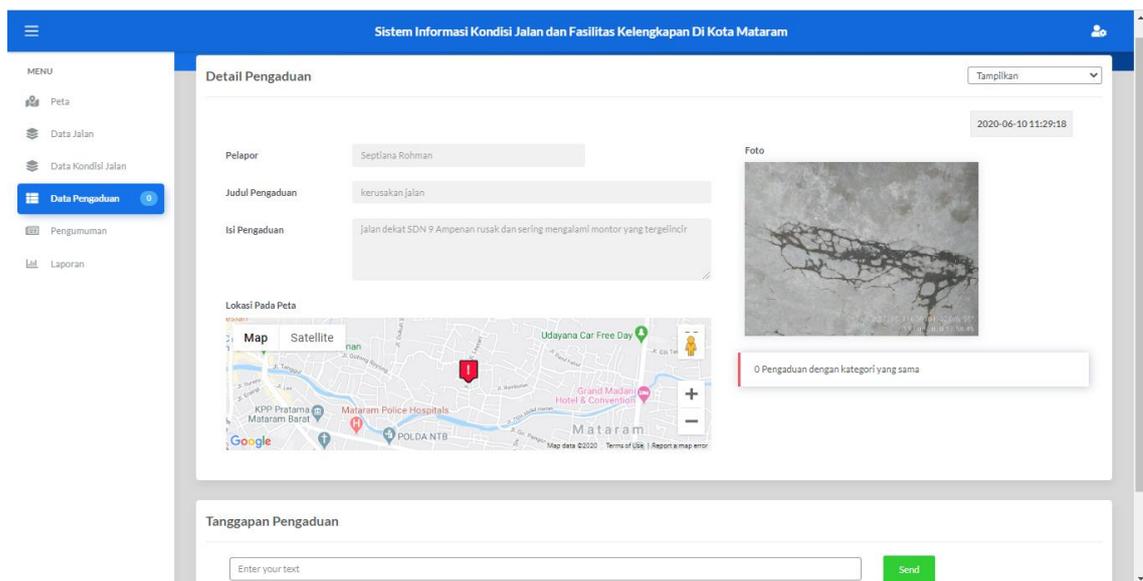
The screenshot shows the 'Data Kondisi Jalan' interface. It features a sidebar menu on the left with options like 'Peta', 'Data Jalan', 'Data Kondisi Jalan', 'Data Pengaduan', 'Pengumuman', and 'Laporan'. The main content area displays a table with columns for 'Nama Jalan', 'Baik', 'Sedang', 'Rusak Ringan', 'Rusak Berat', 'Tahun Data', and 'Action'. The table lists five road entries with their respective counts and years. Below the table, there is a pagination control showing 'Showing 1 to 5 of 5 entries' and 'Previous 1 Next' buttons.

Nama Jalan	Baik	Sedang	Rusak Ringan	Rusak Berat	Tahun Data	Action
Jalan Alpokot	20	7	7	7	2020	[Edit] [Delete]
Jalan Antareja	1.56	0	0	0	2019	[Edit] [Delete]
Jalan Gora 2	1.6	0	0	0	2020	[Edit] [Delete]
Jalan Lalu Mesir	1.02	0.23	0	0	2019	[Edit] [Delete]
Jalan Raden Mas Panji Anom	1	2	1	1	2020	[Edit] [Delete]
Kode Jalan	Baik	Sedang	Rusak Ringan	Rusak Berat	Tahun Data	Action

Gambar 4.11 *Interface* menu data kondisi jalan.

5. Interface detail pengaduan kerusakan jalan

Untuk data pengaduan kerusakan, admin hanya dapat meng-*update* data status pengaduan untuk ditampilkan atau tidak ke pengguna aplikasi *mobile* yang lain dengan memilih *select* pojok kanan atas pada *interface*. Selain itu, *interface* detail pengaduan memiliki *form* untuk melakukan tanggapan pengaduan oleh *admin*. *Interface* menu detail pengaduan kerusakan jalan dapat dilihat pada Gambar 4.12.

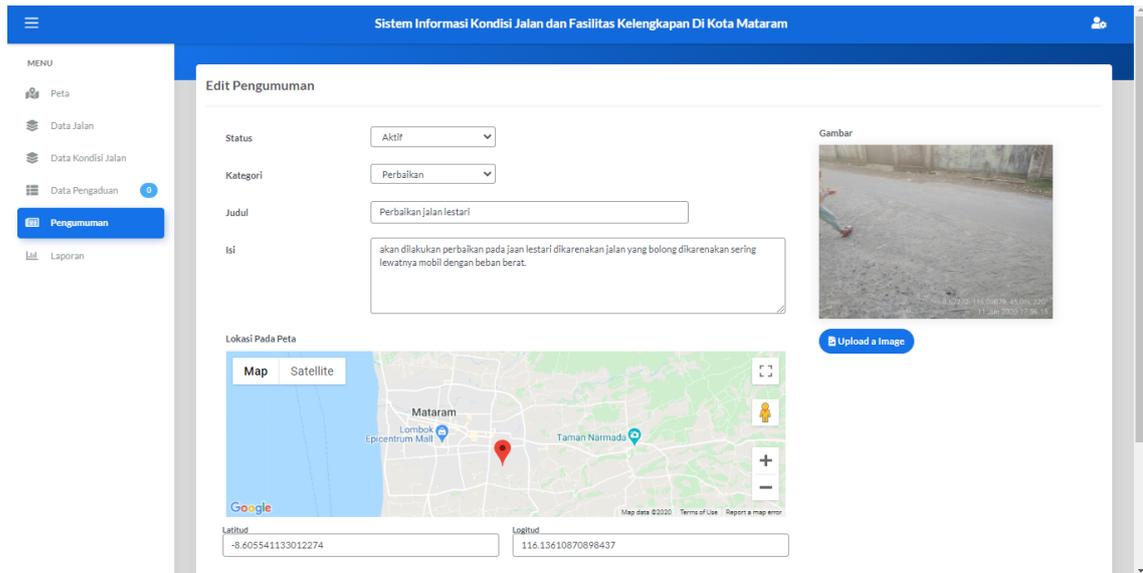


The screenshot shows the 'Detail Pengaduan' interface. It includes a sidebar menu on the left with options like 'Peta', 'Data Jalan', 'Data Kondisi Jalan', 'Data Pengaduan', 'Pengumuman', and 'Laporan'. The main content area displays a form for reporting road damage. The form fields include 'Pelapor' (Septiana Rohman), 'Judul Pengaduan' (kerusakan jalan), and 'Isi Pengaduan' (Jalan dekat SDN 9 Ampenan rusak dan sering mengalami banjir yang tergelincir). There is a 'Foto' section with a photo of a road crack. Below the form is a map showing the location of the report. At the bottom, there is a 'Tanggapan Pengaduan' section with a text input field and a 'Send' button.

Gambar 4.12 *Interface* detail pengaduan kerusakan jalan.

6. Interface edit data pengumuman untuk jalan

Interface tambah data dan edit data pengumuman memiliki struktur *form* yang mirip, perbedaan keduanya adalah *form* edit data pengumuman terdapat *select* status pengumuman untuk menampilkan atau menyembunyikan pengumuman dari pengguna aplikasi *mobile*. *Interface* edit data pengumuman untuk jalan dapat dilihat pada Gambar 4.13.



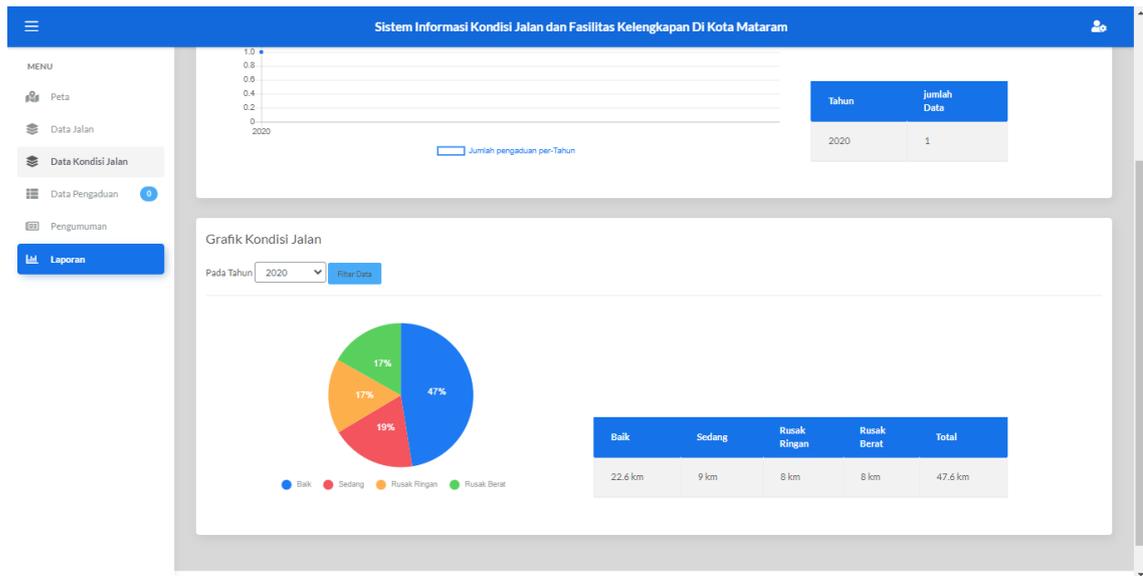
The screenshot shows the 'Edit Pengumuman' form in a web application. The form is titled 'Edit Pengumuman' and is part of the 'Sistem Informasi Kondisi Jalan dan Fasilitas Kelengkapan Di Kota Mataram'. The form includes the following fields and components:

- Status:** A dropdown menu with the value 'Aktif' selected.
- Kategori:** A dropdown menu with the value 'Perbaikan' selected.
- Judul:** A text input field containing 'Perbaikan jalan lestari'.
- Isi:** A text area containing the text 'akan dilakukan perbaikan pada jalan lestari dikarenakan jalan yang bolong dikarenakan sering lewatnya mobil dengan beban berat'.
- Gambar:** A section for uploading an image, showing a preview of a road surface and an 'Upload a Image' button.
- Lokasi Pada Peta:** A map section showing the location in Mataram, with a red pin and a 'Map' button. Below the map, there are input fields for 'Latitude' (-8.605541133012274) and 'Longitude' (116.13610870898437).

Gambar 4.13 *Interface* edit data pengumuman jalan.

7. Interface menu laporan untuk admin jalan

Pada *interface* menu laporan untuk admin jalan terdapat dua buah *grafik* dan tabel. Grafik dan tabel pertama merupakan grafik garis untuk jumlah pengaduan kerusakan jalan pada rentang tahun tertentu, sedangkan grafik dan tabel kedua yaitu grafik lingkaran persentase kondisi jalan pada tahun tertentu. *Interface* menu laporan untuk admin jalan dapat dilihat pada Gambar 4.14.



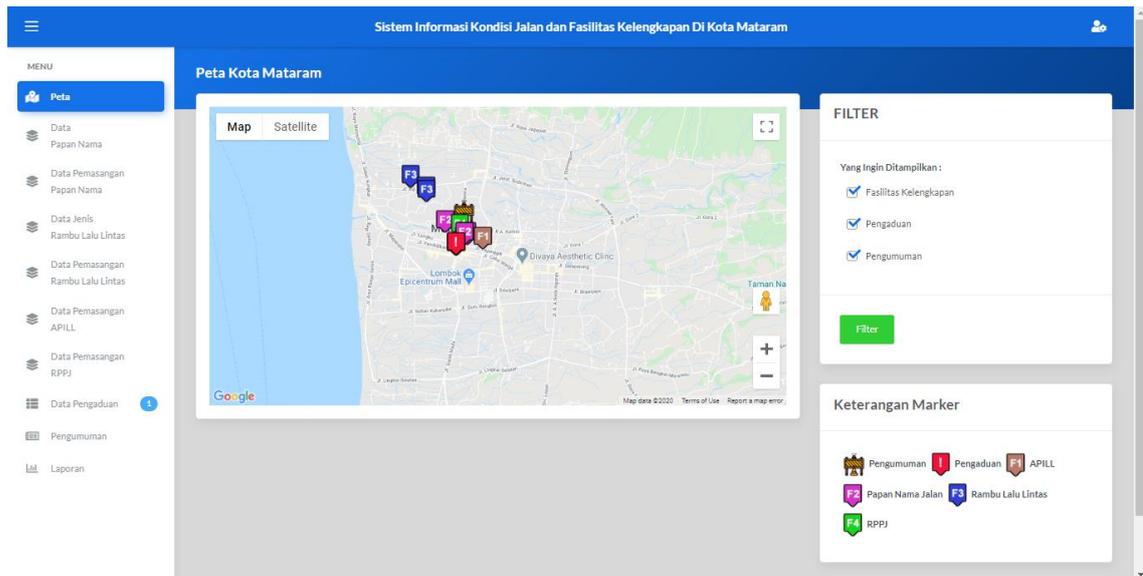
Gambar 4.14 *Interface* menu laporan untuk admin jalan.

4.1.3.2 *Interface* aplikasi website untuk admin fasilitas kelengkapan jalan

Aplikasi *website* untuk admin fasilitas kelengkapan jalan atau admin dari Dinas Perhubungan Kota Mataram terdiri dari beberapa *interface* yang diantaranya adalah sebagai berikut:

1. *Interface* menu peta untuk admin fasilitas kelengkapan

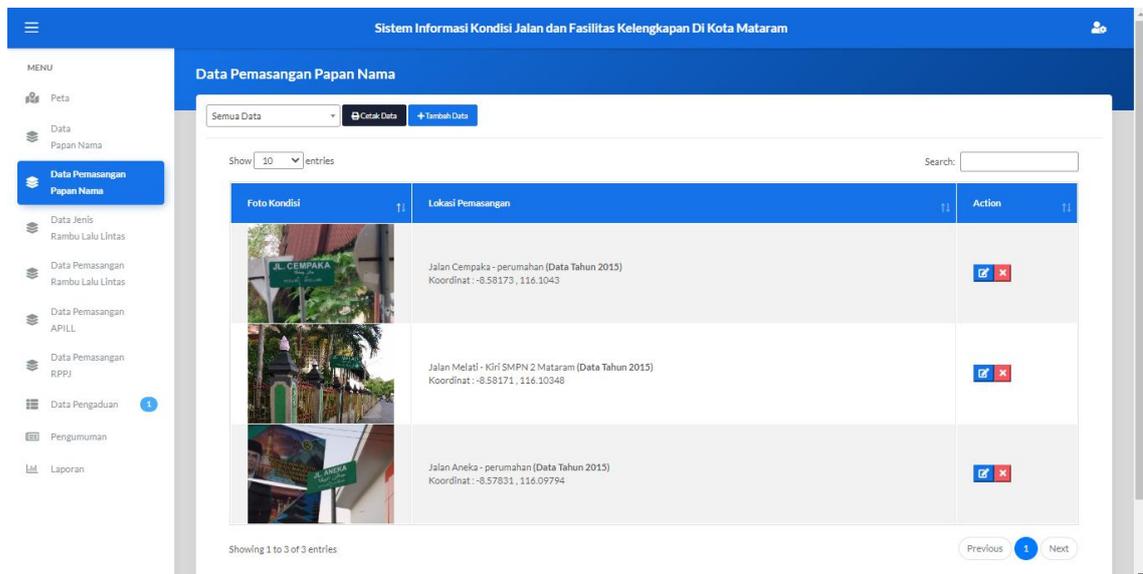
Setelah admin fasilitas kelengkapan berhasil login, maka menu yang pertama kali akan ditampilkan adalah menu Peta. Pada menu ini menampilkan peta yang berpusat di Kota Mataram beserta marker yang merupakan titik koordinat lokasi pemasangan fasilitas kelengkapan jalan, marker yang merupakan titik koordinat lokasi pengaduan kerusakan fasilitas kelengkapan jalan, dan marker yang merupakan titik koordinat lokasi perbaikan/pembangunan fasilitas kelengkapan jalan. *Interface* menu peta untuk admin fasilitas kelengkapan jalan dapat dilihat pada Gambar 4.15.



Gambar 4.15 *Interface* menu peta untuk *admin* fasilitas kelengkapan jalan.

2. Interface menu data pemasangan papan nama jalan

Menu pemasangan papan nama jalan merupakan menu yang hanya dimiliki oleh admin fasilitas kelengkapan jalan. Pada menu ini menampilkan daftar data pemasangan papan nama jalan dalam bentuk tabel yang telah di-*input* oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau edit data dan juga menghapus data pemasangan papan nama jalan. *Interface* menu data pemasangan papan jalan dapat dilihat pada Gambar 4.16.

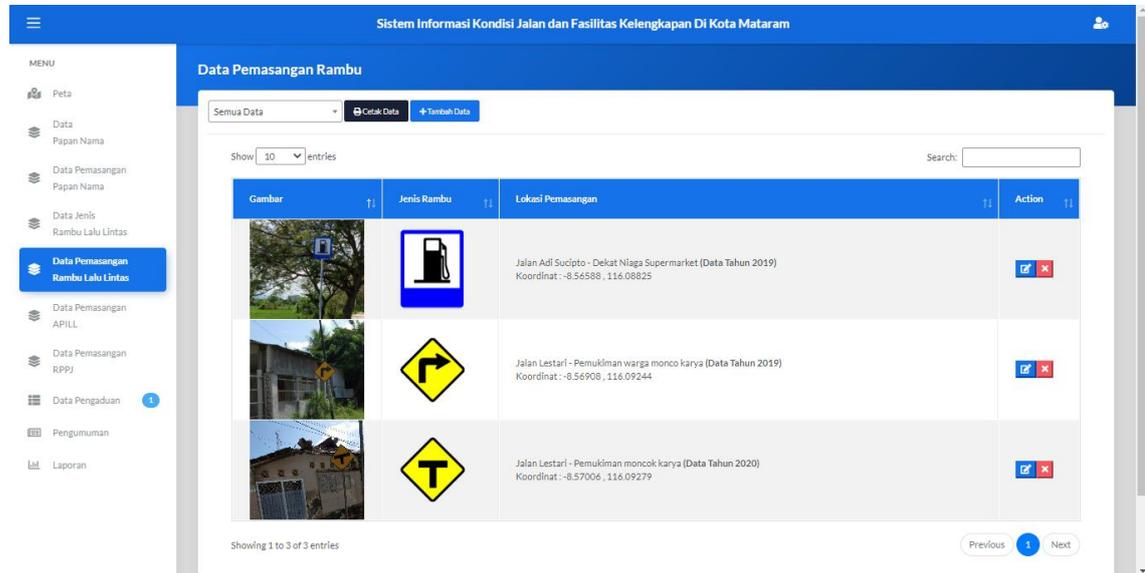


Gambar 4.16 *Interface* menu data pemasangan papan nama jalan.

3. Interface menu data pemasangan rambu lalu lintas

Menu pemasangan rambu lalu lintas merupakan menu yang hanya dimiliki oleh admin fasilitas kelengkapan jalan. Pada menu ini menampilkan daftar data pemasangan

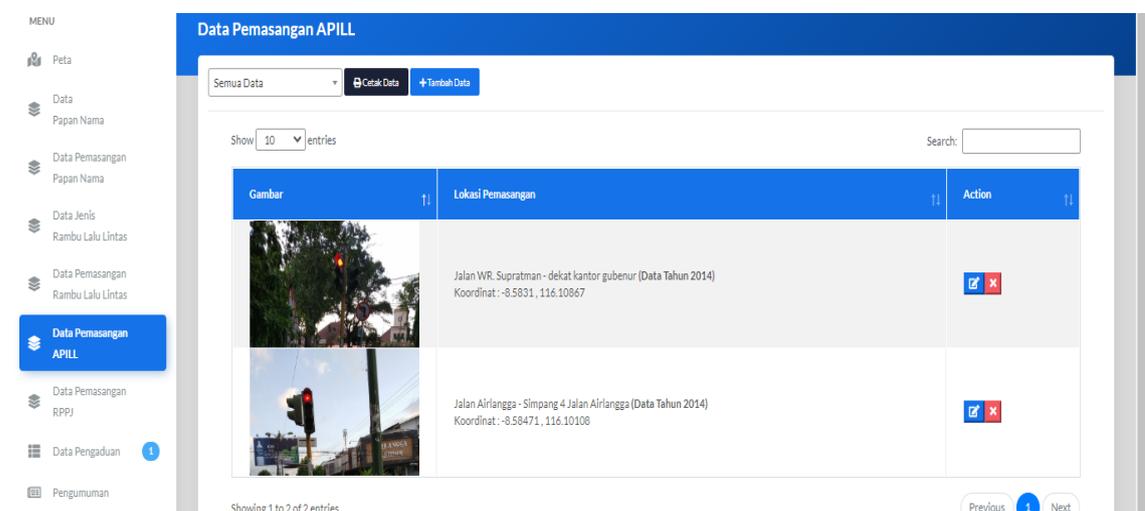
rambu lalu lintas dalam bentuk tabel yang telah di-*input* oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau edit data dan juga menghapus data pemasangan papan nama rambu lalu lintas. *Interface* menu data rambu lalu lintas dapat dilihat pada Gambar 4.17.



Gambar 4.17 *Interface* menu data pemasangan rambu lalu lintas.

4. Interface menu data pemasangan APILL

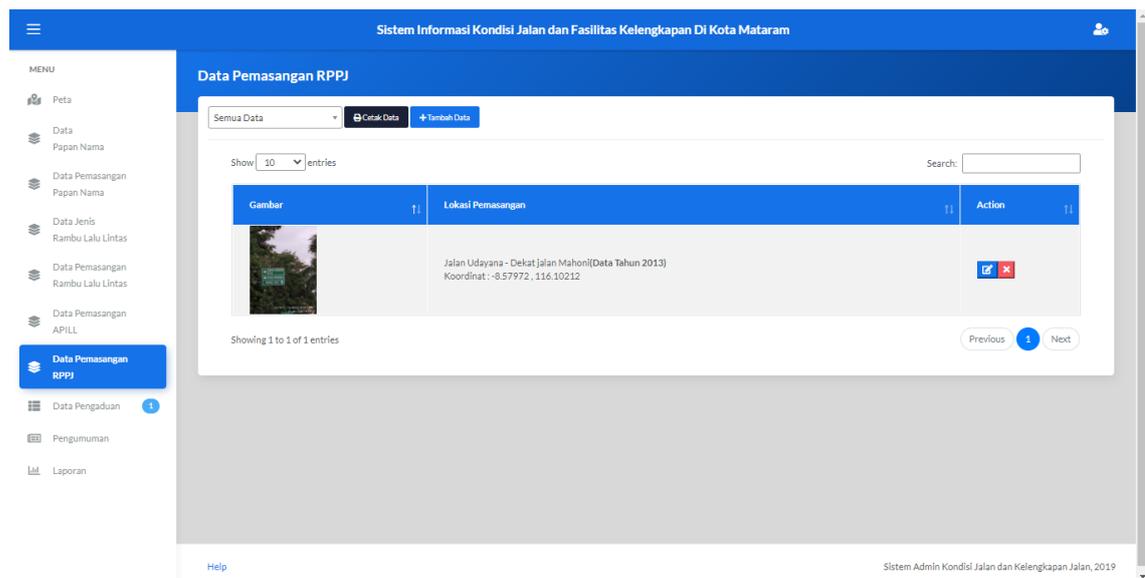
Menu pemasangan APILL atau Alat Pemberi Isyarat Lalu Lintas merupakan menu yang hanya dimiliki oleh admin fasilitas kelengkapan jalan. Pada menu ini menampilkan daftar data pemasangan APILL dalam bentuk tabel yang telah di-*input* oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau edit data dan juga menghapus data pemasangan APILL. *Interface* menu data APILL dapat dilihat pada Gambar 4.18.



Gambar 4.18 *Interface* menu data pemasangan APILL.

5. Interface menu data pemasangan RPPJ

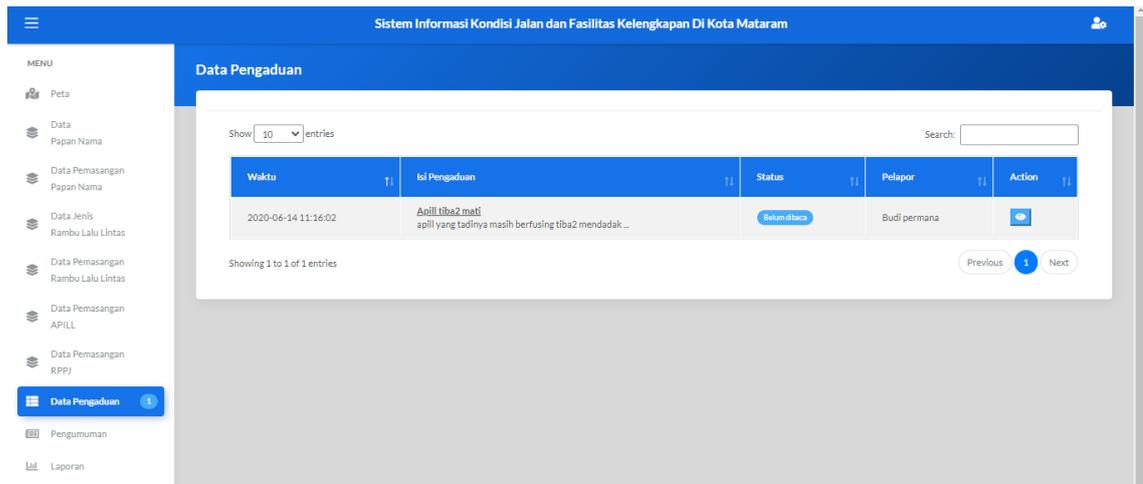
Menu pemasangan RPPJ atau Rambu Pendahulu Petunjuk Jurusan merupakan menu yang hanya dimiliki oleh admin fasilitas kelengkapan jalan. Pada menu ini menampilkan daftar data pemasangan RPPJ dalam bentuk tabel yang telah di-*input* oleh admin sebelumnya. Admin dapat melakukan tambah data, lihat detail data atau edit data dan juga menghapus data pemasangan RPPJ. *Interface* menu data RPPJ dapat dilihat pada Gambar 4.19.



Gambar 4.19 *Interface* menu data pemasangan RPPJ.

6. Interface menu pengaduan kerusakan fasilitas kelengkapan jalan

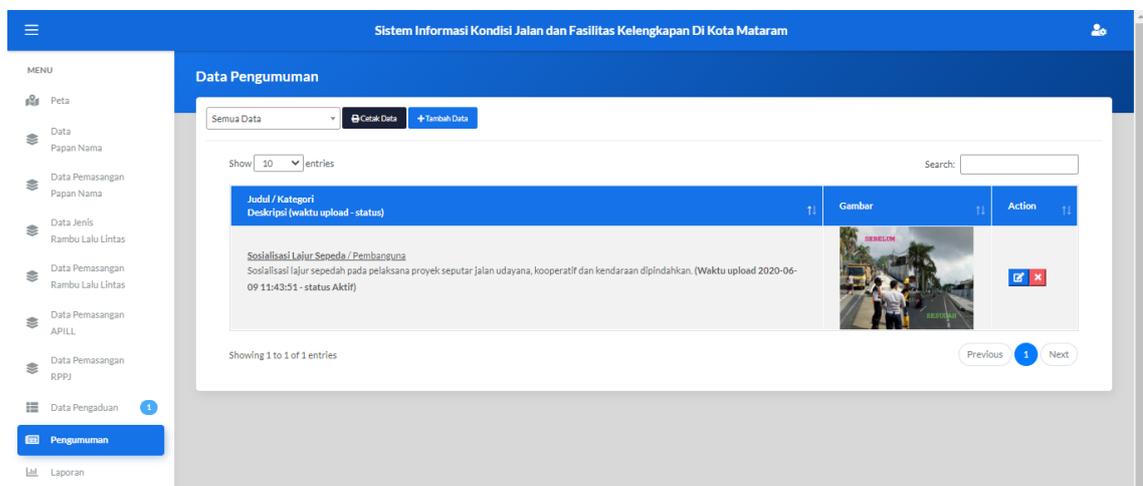
Menu pengaduan yang dimiliki oleh admin fasilitas kelengkapan jalan ini menampilkan daftar data pengaduan kerusakan fasilitas kelengkapan jalan dalam bentuk tabel. Admin dapat melihat detail pengaduan, kemudian memberi tanggapan pengaduan dan meng-*update* status pengaduan untuk diperlihatkan atau tidak ke penggunaan aplikasi *mobile*. *Interface* menu menu pengaduan kerusakan fasilitas kelengkapan dapat dilihat pada Gambar 4.20.



Gambar 4.20 *Interface* menu pengaduan kerusakan fasilitas kelengkapan jalan.

7. Interface menu pengumuman fasilitas kelengkapan jalan

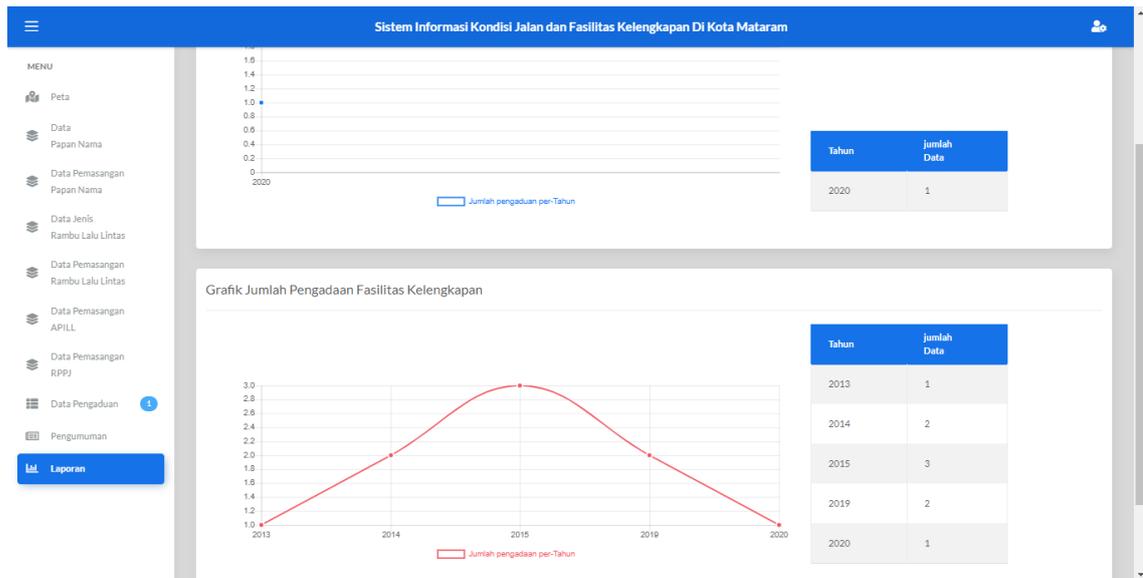
Menu pengumuman yang dimiliki oleh admin fasilitas kelengkapan jalan ini menampilkan daftar data pengumuman perbaikan/pembangunan fasilitas kelengkapan jalan dalam bentuk tabel. Admin dapat melakukan tambah data, edit data, dan hapus data pengumuman. *Interface* menu pengumuman fasilitas kelengkapan jalan dapat dilihat pada Gambar 4.21.



Gambar 4.21 *Interface* menu pengumuman fasilitas kelengkapan jalan.

8. Interface menu laporan untuk admin fasilitas kelengkapan jalan

Pada *interface* menu laporan untuk admin fasilitas kelengkapan jalan terdapat dua buah *grafik* dan tabel. *Grafik* dan tabel pertama merupakan *grafik* garis untuk jumlah pengaduan kerusakan fasilitas kelengkapan jalan pada rentang tahun tertentu, sedangkan *grafik* dan tabel kedua yaitu *grafik* garis untuk jumlah pemasangan fasilitas kelengkapan jalan pada rentang tahun tertentu. *Interface* menu laporan untuk admin fasilitas kelengkapan jalan dapat dilihat pada Gambar 4.22.



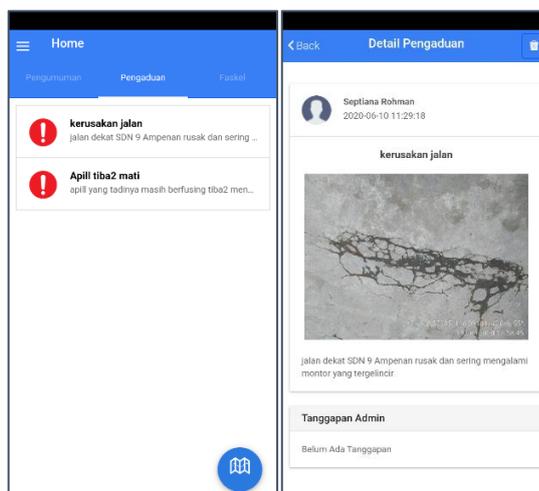
Gambar 4.22 *Interface* menu laporan untuk admin fasilitas kelengkapan jalan.

4.1.3.3 *Interface* pada aplikasi *mobile*

Aplikasi *mobile* untuk masyarakat Kota Mataram terdiri dari beberapa *interface* yang diantaranya adalah sebagai berikut:

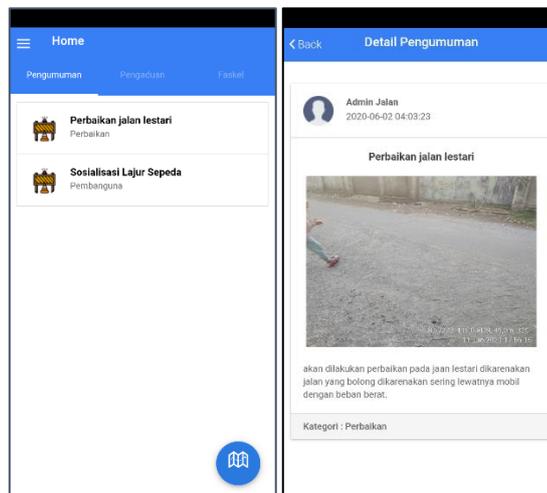
1. *Interface* menu *home* - list data

Setelah berhasil login menu yang pertama ditampilkan adalah menu home-list data. Pada menu home terdapat 3 buah tab yaitu tab pengumuman berisi daftar pengumuman yang diberikan oleh admin jalan maupun admin fasilitas kelengkapan jalan, tab pengaduan berisi daftar pengaduan yang telah dilakukan oleh masyarakat Kota Mataram, dan tab faskel berisi daftar fasilitas kelengkapan jalan yang terpasang di Kota Mataram.



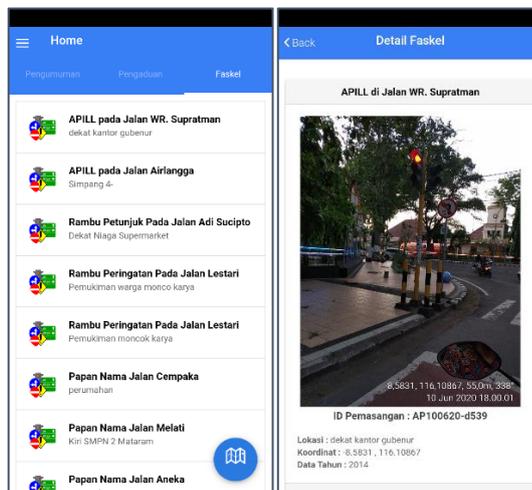
Gambar 4.23 *Interface* menu home untuk tab *list* pengaduan (kiri) dan detail pengaduan(kanan). Pada Gambar 4.23 merupakan *interface* menu *home* untuk tab pengaduan yang merupakan tampilan pertama kali yang muncul saat berhasil *login*(kiri), dengan mengklik

salah satu data dari list tab pengaduan maka akan memunculkan detail data dari pengaduan tersebut(kanan).



Gambar 4.24 *Interface* menu home untuk tab *list* pengumuman (kiri) dan detail pengumuman(kanan).

Pada Gambar 4.24 merupakan *interface* menu *home* untuk tab *list* pengumuman yang berisi daftar pengumuman yang dilakukan oleh *admin*(kiri), dengan mengklik salah satu data dari list tab pengumuman maka akan memunculkan detail data dari pengumuman tersebut(kanan).

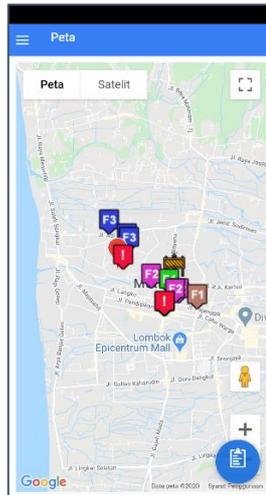


Gambar 4.25 *Interface* menu home untuk tab *list* faskel (kiri) dan detail faskel(kanan).

Pada Gambar 4.25 merupakan *interface* menu *home* untuk tab *list* faskel yang berisi daftar fasilitas kelengkapan jalan yang terpasang di kota Mataram(kiri), dengan mengklik salah satu data dari *list* tab faskel maka akan memunculkan detail data dari fasilitas kelengkapan jalan tersebut(kanan).

2. Interface menu home – peta

Pada menu home – list data terdapat tombol pada pojok kanan bawah untuk mengarahkan ke tampilan peta yang berpusat di kota Mataram yang berisi marker-marker titik koordinat posisi pengaduan kerusakan, titik koordinat posisi pengumuman, dan titik koordinat posisi pemasangan fasilitas kelengkapan jalan. Interface menu home – peta dapat dilihat pada Gambar 4.26.



Gambar 4.26 *Interface* menu home-peta.

3. Interface untuk melakukan pengaduan

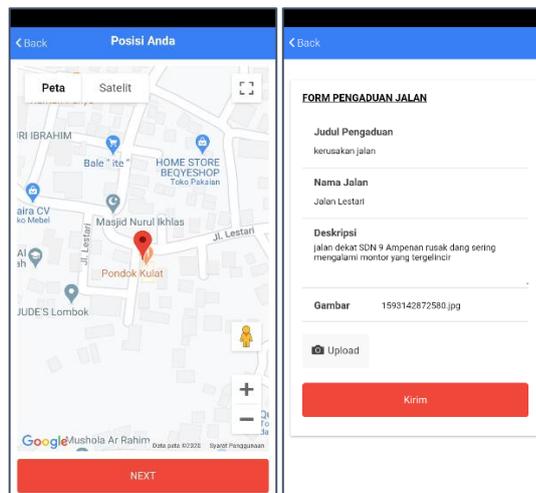
Untuk melakukan pengaduan dengan memilih menu pengaduan maka akan mengarahkan ke tampilan kategori pengaduan yang dapat dilihat pada Gambar 4.27.



Gambar 4.27 kategori pengaduan kerusakan.

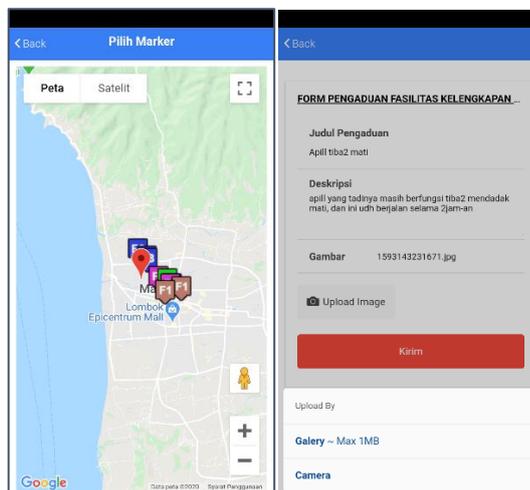
Pada Gambar 4.27 merupakan *interface* kategori pengaduan kerusakan, dimana saat memilih menu pengaduan jalan maka akan menampilkan peta dengan pusat posisi pengguna aplikasi *mobile* berada kemudia memiliki tombol *next* yang mengarahkan ke

halaman form pengaduan jalan. Tampilan peta dan form pengaduan kerusakan jalan dapat dilihat pada Gambar 4.28.



Gambar 4.28 *Interface* peta (kiri) dan *form* penagduan kerusakan jalan (kanan).

Pada Gambar 4.27 merupakan *interface* kategori pengaduan kerusakan, dimana saat memilih menu pengaduan fasilitas kelengkapan maka akan menampilkan peta dengan marker-marker posisi pemasangan fasilitas kelengkapan jalan yang dapat dipilih untuk dilaporkan kemudia akan mengarahkan ke halaman form pengaduan fasilitas kelengkapan jalan. Tampilan peta dan form pengaduan kerusakan fasilitas kelengkapan jalan dapat dilihat pada Gambar 4.29.



Gambar 4.29 *Interface* peta (kiri) dan *form* penagduan kerusakan fasilitas kelengkapan jalan (kanan).

4. Interface menu pengaduanku

Pada menu pengaduanku terdapat list data pengaduan yang telah dilakukan pemilik akun sebelumnya. Pada list dapat dilihat status dari pengaduan yang telah dilakukan apakah sudah dibaca (status diterima) oleh admin yang bersangkutan dengan kategori

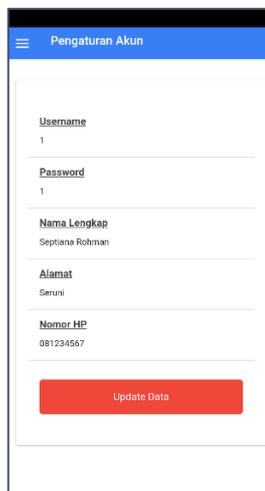
yang kita adukan kerusakannya atau belum (status dikirim). Interface menu pengaduanku dapat dilihat pada Gambar 4.30.



Gambar 4.30 *Interface* menu pengaduanku.

5. Interface menu pengaturan akun

Pada menu pengaturan akun menampilkan *form* yang berisi data dari pemilik akun yang sesuai dengan data saat membuat akun, data akun tersebut dapat diganti sesuai keinginan pemilik akun. Interface menu pengaturan akun dapat dilihat pada Gambar 4.31.



Gambar 4.31 *Interface* menu pengaturan akun.

4.2 Pengujian Sistem

Pengujian sistem dilakukan setelah melalui tahap implementasi untuk mengetahui sejauh mana keberhasilan sistem dan kualitasnya. Pengujian sistem yang dilakukan antara lain dengan menggunakan metode pengujian *black box* dan juga *mean opinion score* (MOS).

4.2.1 Hasil Pengujian Metode *Black Box*

Metode pengujian *black box* merupakan pengujian yang menguji fungsi-fungsi di dalam sistem untuk menentukan apakah fungsi-fungsi tersebut sudah berjalan sesuai harapan atau tidak. Pengujian *black box* dilakukan pada saat pengguna aplikasi *web* ataupun *mobile* melakukan percobaan pada sistem. Adapun hasil pengujian *black box* secara lengkap dapat dilihat pada lampiran III. Berikut merupakan jabaran fungsi-fungsi di dalam sistem yang akan di uji menggunakan metode *black box*.

4.2.1.1 Pengujian aplikasi untuk admin

Pengujian aplikasi web untuk admin menggunakan metode *black box* di lakukan pada beberapa fitur yaitu login, mengelola data jalan, mengelola kondisi jalan, mengelola data papan nama jalan, mengelola data pemasangan papan nama jalan, mengelola data jenis rambu, mengelola data pemasangan rambu, mengelola data pemasangan apill, mengelola data pemasangan rppj, mengelola pengumuman, mengelola data pengaduan, memberikan tanggapan pengaduan, dan melihat grafik tahunan.

a. Pengujian mengelola pengumuman

Pengujian mengelola pengumuman dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah pengumuman sudah berjalan dengan baik dan benar. Pengujian mengelola pengumuman akan dijabarkan sebagai berikut:

- Fungsi tambah pengumuman

Fungsi tambah pengumuman merupakan fungsi untuk menambahkan pengumuman ke database sistem yang dilakukan oleh admin jalan maupun admin faskel. Pengujian fungsi tambah pengumuman akan dijelaskan pada Tabel 4.1.

Tabel 4.1 Pengujian fungsi tambah pengumuman.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah pengumuman	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid

Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu pengumuman dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid
--	---	--------	-------

- Fungsi edit pengumuman

Fungsi edit pengumuman merupakan fungsi untuk merubah data pengumuman untuk memperbaiki data sebelumnya yang dilakukan oleh admin jalan maupun admin faskel. Pengujian fungsi edit pengumuman akan dijelaskan pada Tabel 4.2.

Tabel 4.2 Pengujian fungsi edit pengumuman.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit pengumuman, beserta menampilkan data yang akan dirubah ke <i>field</i> pada form.	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian mengarahkan ke halaman menu pengumuman dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus pengumuman

Fungsi hapus pengumuman merupakan fungsi untuk menghapus data pengumuman pada database jalan yang dilakukan oleh admin jalan maupun admin faskel. Pengujian fungsi hapus pengumuman akan dijelaskan pada Tabel 4.3.

Tabel 4.3 Pengujian fungsi hapus pengumuman.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Yes	Data akan dihapus dari <i>database</i> dan tidak terlihat lagi dalam tabel pengumuman.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Cancel	Data tidak akan dihapus dari <i>database</i> dan masih terlihat dalam tabel pengumuman.	Sesuai	Valid

b. Pengujian mengelola pengaduan

Pengujian mengelola pengaduan dilakukan untuk menguji apakah fungsi untuk melihat pengaduan dan melakukan tanggapan pengaduan sudah berjalan dengan baik dan benar. Pengujian mengelola pengaduan akan dijelaskan pada Tabel 4.4.

Tabel 4.4 Pengujian fungsi hapus pengumuman.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Menakn tombol detail salah satu pengaduan	Mengarakan ke halaman detail pengaduan yang dipilih.	Sesuai	Valid
Mengisi <i>form</i> untuk tanggapan pengaduan, kemudian menekan tombol Kirim.	Menampilkan tanggapan yang telah dilakukan pada bagian bawah detail pengaduan.	Sesuai	Valid

4.2.1.2 Pengujian aplikasi untuk masyarakat

Pengujian aplikasi *mobile* untuk masyarakat menggunakan metode *black box* di lakukan pada beberapa fitur yaitu *login*, *sign up*, melihat kondisi jalan dan kelengkapan jalan, dan mengirim pengaduan kerusakan.

a. Pengujian melihat kondisi jalan dan kelengkapan jalan

Pengujian melihat kondisi jalan dan kelengkapan jalan dilakukan untuk menguji apakah fungsi untuk melihat fasilitas kelengkapan jalan, melihat pengumuman yang dilakukan *admin*, dan melihat pengaduan yang telah dilakukan masyarakat sudah berjalan dengan baik dan benar. Pengujian melihat kondisi jalan dan kelengkapan jalan akan dijabarkan sebagai berikut:

- Fungsi melihat fasilitas kelengkapan jalan

Fungsi melihat fasilitas kelengkapan jalan merupakan fungsi untuk melihat fasilitas-fasilitas kelengkapan jalan yang terpasang di Kota Mataram. Pengujian fungsi melihat fasilitas kelengkapan jalan akan dijelaskan pada Tabel 4.5.

Tabel 4.5 Pengujian fungsi melihat fasilitas kelengkapan jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Pada menu home memilih tab Faskel	Muncul <i>list</i> dari fasilitas-fasilitas kelengkapan jalan berdasarkan data dari dinas.	Sesuai	Valid

Mengklik salah satu dari <i>list</i> fasilitas kelengkapan jalan.	Menampilkan halaman detail dari fasilitas kelengkapan yang dipilih.	Sesuai	Valid
---	---	--------	-------

- Fungsi melihat pengumuman

Fungsi melihat pengumuman merupakan fungsi untuk melihat pengumuman perbaikan dan pembangunan dari jalan maupun fasilitas kelengkapan jalan di Kota Mataram. Pengujian fungsi melihat pengumuman akan dijelaskan pada Tabel 4.6.

Tabel 4.6 Pengujian fungsi melihat pengumuman.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Pada menu home memilih tab Pengumuman.	Muncul <i>list</i> dari pengumuman berdasarkan data dari dinas.	Sesuai	Valid
Mengklik salah satu dari <i>list</i> pengumuman.	Menampilkan halaman detail dari pengumuman yang dipilih.	Sesuai	Valid

- Fungsi melihat pengaduan

Fungsi melihat pengaduan merupakan fungsi untuk melihat pengaduan kerusakan jalan maupun fasilitas kelengkapan jalan yang dilakukan oleh masyarakat di Kota Mataram. Pengujian fungsi melihat pengaduan akan dijelaskan pada Tabel 4.7.

Tabel 4.7 Pengujian fungsi melihat pengaduan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Pada menu home memilih tab Pengaduan.	Muncul <i>list</i> dari pengaduan yang telah dilakukan oleh masyarakat melalui aplikasi <i>mobile</i> .	Sesuai	Valid
Mengklik salah satu dari <i>list</i> pengaduan.	Menampilkan halaman detail dari pengaduan yang dipilih beserta tanggapan yang dilakukan oleh dinas.	Sesuai	Valid

- b. Pengujian melakukan pengaduan kerusakan

Pengujian melakukan pengaduan kerusakan dilakukan untuk menguji apakah fungsi untuk melakukan pengaduan kerusakan sudah berjalan dengan baik dan benar. Masyarakat akan mengisi *form* dan mengirimkan pengaduan kerusakan jalan maupun

fasilitas kelengkapan jalan ke Dinas. Pengujian melakukan pengaduan kerusakan akan dijelaskan pada Tabel 4.8.

Tabel 4.8 Pengujian melakukan pengaduan kerusakan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Pada menu Pengaduan memilih kategori pengaduan.	Mengarahkan ke halaman peta untuk memastika koordinat pengadu.	Sesuai	Valid
Setelah itu memilih tombol next atau lapor yang ada di halaman peta.	Mengarahkan ke <i>form</i> pengaduan dengan <i>field</i> koordinat telah terisi.	Sesuai	Valid
Mengisi <i>field</i> yang masih kosong pada <i>form</i> kemudia menekan tombol kirim	Pengaduan berhasil dikirim dan memunculkan pesan berhasil melakukan pengaduan.	Sesuai	Valid

4.2.2 Hasil Pengujian Mean Opinion Score (MOS)

Pengujian MOS digunakan untuk mengukur kesesuaian aplikasi berdasarkan pendapat subjektif dari pengguna sistem. Pengujian MOS dilakukan dengan cara mendemokan sistem secara langsung kepada responden, lalu kemudian responden mengisi kuesioner dengan beberapa pernyataan. Jumlah keseluruhan responden sebanyak 30 orang, yang terdiri dari 3 orang dari Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram, 3 orang dari Dinas Perhubungan Kota Mataram, serta 24 orang dari masyarakat Kota Mataram. Dari setiap pernyataan yang diberikan, responden akan memberikan penilaian dengan keterangan sebagai berikut:

1. Sangat Setuju (SS)
2. Setuju (S)
3. Cukup (C)
4. Tidak Setuju (TS)
5. Sangat Tidak Setuju (STS)

Selanjutnya akan dihitung persentase penilaian yang diberikan responden di setiap pernyataan. Rumus untuk menghitung persentase nilai adalah sebagai berikut:

$$\text{Persentase} = J / N * 100\%$$

Keterangan:

J: Total nilai jawaban responden pada setiap nomor.

N: Jumlah Respoden.

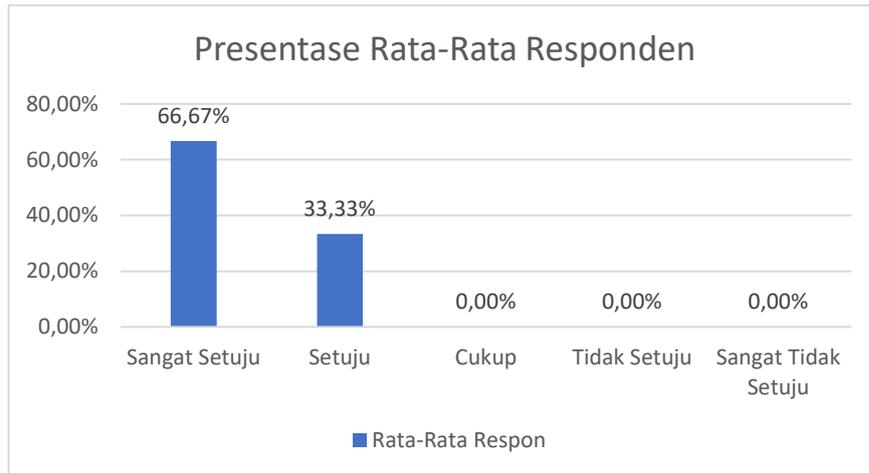
Kuesioner pengujian sistem yang diberikan untuk Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram terdiri dari 4 pernyataan, yaitu sebagai berikut:

1. Sistem dapat mempermudah melakukan pendataan jalan dan kondisi jalan setiap tahunnya.
2. Sistem dapat mempermudah menginformasikan tentang adanya kegiatan perbaikan/pembangunan jalan.
3. Sistem mempermudah menerima pengaduan kerusakan oleh masyarakat.
4. Sistem dapat mempermudah melakukan tanggapan untuk pengaduan yang dilakukan.

Tabel 4.9 hasil pengujian MOS Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.

Pernyataan	Kategori penilaian					Jumlah Responden	Presentase (%)				
	SS	S	C	TS	STS		SS	S	C	TS	STS
1	3	0	0	0	0	3	100	0.00	0.00	0.00	0.00
2	3	0	0	0	0	3	100	0.00	0.00	0.00	0.00
3	1	2	0	0	0	3	33,33	66,67	0.00	0.00	0.00
4	1	2	0	0	0	3	33,33	66,67	0.00	0.00	0.00
Rata-rata							66,67	33,33	0.00	0.00	0.00

Berdasarkan hasil pengujian MOS pada Tabel 4.9, maka dapat dihitung total keseluruhan dari pernyataan responden yaitu dengan jumlah rata-rata penilaian sangat setuju oleh responden yaitu 66,67%, dan dengan jumlah rata-rata penilaian setuju oleh responden yaitu 33,33%. Adapun grafik jumlah presentase pernyataan rata-rata responden dari Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram dari kuesioner yang telah diajukan dapat dilihat pada Gambar 4.32.



Gambar 4.32 Grafik rata-rata responden Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.

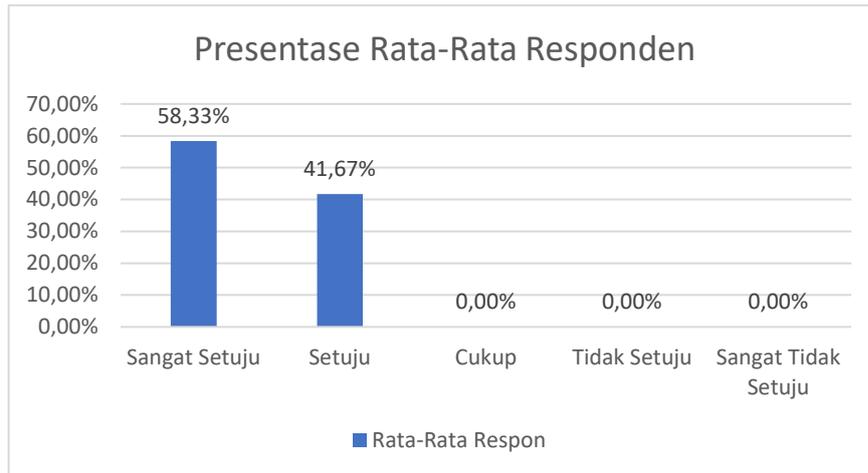
Kuesioner pengujian sistem yang diberikan untuk Dinas Perhubungan Kota Mataram terdiri dari 4 pernyataan, yaitu sebagai berikut:

1. Sistem dapat mempermudah melakukan pendataan pemasangan fasilitas kelengkapan jalan (Papan Nama Jalan, Rambu Lalu Lintas, APILL, dan RPPJ).
2. Sistem dapat mempermudah menginformasikan tentang adanya kegiatan perbaikan/pembangunan fasilitas kelengkapan jalan.
3. Sistem mempermudah menerima pengaduan kerusakan oleh masyarakat.
4. Sistem dapat mempermudah melakukan tanggapan untuk pengaduan yang dilakukan.

Tabel 4.10 hasil pengujian MOS Dinas Perhubungan Kota Mataram.

Pernyataan	Kategori penilaian					Jumlah Responden	Presentase (%)				
	SS	S	C	TS	STS		SS	S	C	TS	STS
1	2	1	0	0	0	3	66,67	33,33	0.00	0.00	0.00
2	2	1	0	0	0	3	66,67	33,33	0.00	0.00	0.00
3	2	1	0	0	0	3	66,67	33,33	0.00	0.00	0.00
4	1	2	0	0	0	3	33,33	66,67	0.00	0.00	0.00
Rata-rata							58,33	41,67	0.00	0.00	0.00

Berdasarkan hasil pengujian MOS pada Tabel 4.10, maka dapat dihitung total keseluruhan dari pernyataan responden yaitu dengan jumlah rata-rata penilaian sangat setuju oleh responden yaitu 58,33%, dan dengan jumlah rata-rata penilaian setuju oleh responden yaitu 41,67%. Adapun grafik jumlah presentase pernyataan rata-rata responden dari Dinas Perhubungan Kota Mataram dari kuesioner yang telah diajukan dapat dilihat pada Gambar 4.33.



Gambar 4.33 Grafik rata-rata responden Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram.

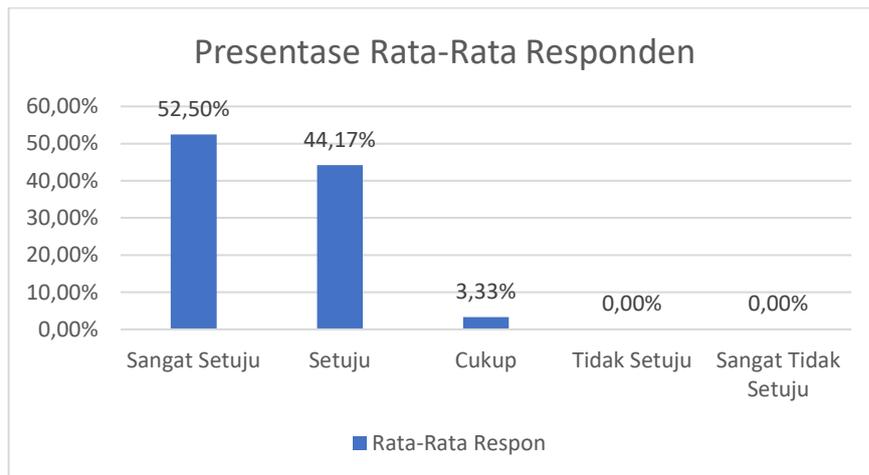
Kuesioner pengujian sistem yang diberikan untuk Masyarakat di Kota Mataram terdiri dari 5 pernyataan, yaitu sebagai berikut:

1. Sistem mempermudah masyarakat melakukan pengaduan kerusakan jalan ataupun kerusakan fasilitas kelengkapan jalan.
2. Sistem mempermudah masyarakat melihat respon terhadap pengaduan yang telah dilakukan sebelumnya.
3. Sistem dapat memperlihatkan pengaduan kerusakan yang dilakukan oleh masyarakat yang lainnya.
4. Sistem mempermudah untuk mengetahui informasi tentang perbaikan/pembangunan jalan dan fasilitas kelengkapan jalan yang dilakukan oleh pihak yang bertanggung jawab.
5. Sistem dapat memperlihatkan posisi pengaduan, posisi perbaikan/pembangunan, posisi pemasangan fasilitas kelengkapan jalan di kota Mataram dalam bentuk peta.

Tabel 4.11 hasil pengujian MOS masyarakat Kota Mataram.

Pernyataan	Kategori penilaian					Jumlah Responden	Presentase (%)				
	SS	S	C	TS	STS		SS	S	C	TS	STS
1	19	5	0	0	0	24	79,17	20,83	0,00	0,00	0,00
2	10	14	0	0	0	24	41,67	58,33	0,00	0,00	0,00
3	9	14	1	0	0	24	37,50	58,33	4,17	0,00	0,00
4	13	9	2	0	0	24	54,17	37,50	8,33	0,00	0,00
5	12	11	1	0	0	24	50,00	45,83	4,17	0,00	0,00
Rata-rata							52,50	44,17	3,33	0,00	0,00

Berdasarkan hasil pengujian MOS pada Tabel 4.11, maka dapat dihitung total keseluruhan dari pernyataan responden yaitu dengan jumlah rata-rata penilaian sangat setuju oleh responden yaitu 52,50%, dengan jumlah rata-rata penilaian setuju oleh responden yaitu 44,17%, dan dengan jumlah rata-rata penilaian cukup oleh responden yaitu 3,33%. Adapun grafik jumlah presentase pernyataan rata-rata responden dari masyarakat Kota Mataram dari kuesioner yang telah diajukan dapat dilihat pada Gambar 4.34.



Gambar 4.34 Grafik rata-rata responden masyarakat.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan pembangunan aplikasi yang dilakukan, terdapat beberapa hal yang dapat disimpulkan, antara lain:

1. Aplikasi *web* yang telah dibuat memudahkan mengelola data jalan dan fasilitas kelengkapan jalan, memudahkan menyampaikaian informasi perbaikan/pembangunan jalan maupun fasilitas kelengkapan jalan di Kota Mataram, serta dapat menerima pengaduan kerusakan dari masyarakat.
2. Aplikasi *mobile* yang telah dibuat mempermudah masyarakat melihat kondisi jalan dan fasilitas kelengkapan jalan yang ada di Kota Mataram, serta dapat melakukan pengaduan kerusakan jalan dan fasilitas kelengkapan jalan kepada dinas yang bertanggung jawab.
3. Dari hasil pengujian *black box* yang telah dilakukan menunjukkan bahwa seluruh fungsi-fungsi menu pada aplikasi yang sudah dibangun sesuai dengan sistem yang diusulkan.
4. Berdasarkan hasil pengujian *Mean Opinion Score* (MOS) yang telah dilakukan menunjukkan bahwa rata-rata penilaian responden dari Dinas Pekerjaan Umum dan Penataan Ruang Kota Mataram terhadap aplikasi *web* menyatakan sangat setuju dan setuju dengan persentase masing-masing 66,67% dan 33,33%, sedangkan rata-rata penilaian responden dari Dinas Perhubungan Kota Mataram terhadap aplikasi *web* menyatakan sangat setuju dan setuju dengan persentase masing-masing 58,33% dan 41,67%, dan rata-rata penilaian responden dari masyarakat terhadap aplikasi *mobile* menyatakan sangat setuju dan setuju dengan persentasi masing-masing 52,50% dan 44,17%. Sehingga dapat disimpulkan bahwa sistem ini layak untuk digunakan.

5.2 Saran

Setelah merancang dan membangun Aplikasi Hybrid Kondisi Jalan dan Fasilitas Kelengkapan Jalan di Kota Mataram ini, ada beberapa saran yang akan disampaikan untuk pengembangan lebih lanjut, yaitu:

1. Untuk penelitian selanjutnya, peneliti bisa mengembangkan dengan penambahan aplikasi *mobile* yang dapat digunakan oleh pegawai dari dinas yang bertugas di

lapangan agar memudahkan dalam pemantau dan pendataan kondisi jalan dan fasilitas kelengkapan jalan secara cepat dan langung.

2. Untuk penelitian selanjutnya, peneliti dapat menambahkan fitur untuk notifikasi data/informasi baru pada aplikasi *mobile* yang digunakan oleh masyarakat.
3. Untuk penelitian selanjutnya, peneliti dapat lebih meningkatkan keamanan sistem dengan menambahkan fitur untuk verifikasi sistem berdasarkan email atau dengan nomor yang dapat dihubungi sehingga dapat lebih memudahkan dalam pengamanan akun.
4. Untuk penelitian selanjutnya, peneliti dapat melakukan migrasi data atau mengevaluasi fitur untuk pengembangan sistem dan kemudian melakukan pengujian dengan metode tertentu seperti *User Acceptance Test*.

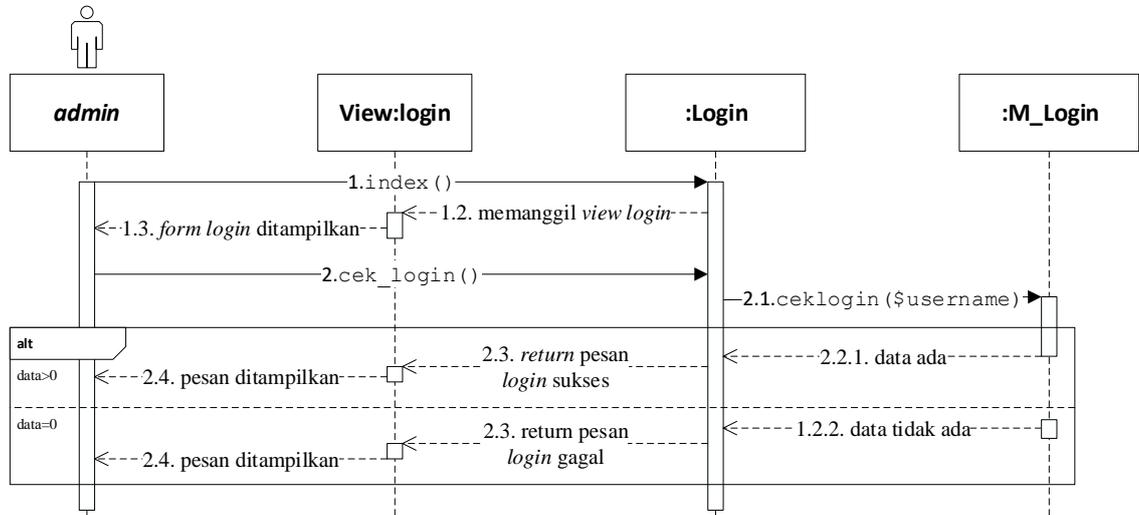
DAFTAR PUSTAKA

- [1] Republik Indonesia, “*Peraturan Menteri Pekerjaan Umum Nomor 13 Tahun 2011 tentang Tata Cara Pemeliharaan dan Penilikan Jalan*”, Jakarta: Menteri Pekerjaan Umum, 2011.
- [2] M. G. Nastiar, “Sistem Pelaporan Kerusakan Jalan Pada Bina Marga Makassar Berbasis Web dan Mobile” Skripsi pada Program Studi Sains dan Teknologi, UIN Alauddin Makassar, Makassar, 2016.
- [3] D. O. Sihombing, “Perancangan Sistem Informasi Pelaporan Masyarakat Untuk Kerusakan Jalan Di Pontianak Menggunakan Google Maps API”, *Jurnal Khatulistiwa Informatika*, vol. 4, no. 1, 2016.
- [4] A. B. A. Pramuji, “Aplikasi Crowdsourcing Pelaporan Kerusakan Jalan Di Dinas Pekerjaan Umum dan Perumahan Sleman Berbasis Android”, Skripsi pada Program Studi Sistem Informasi, Sekolah Tinggi Manajemen Informatika dan Komputer, AKAKOM, Yogyakarta, 2017.
- [5] A. P. Basuki, “Proyek Membangun Website Berbasis PHP dengan CodeIgniter”, Yogyakarta: Lokomedia, 2014.
- [6] J. F. Tompoh, S. R. Sentinuwo, dan A. A. E. Sinsuw, “Rancang Bangun Aplikasi Pemesanan Menu Makanan Restoran Berbasis Android”, *E-jurnal Teknik Informatika*, Vol. 9, No. 1, 2016.
- [7] Rosa dan M. Shalahuddin, “*Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*”, Bandung: Informatika, 2013.
- [8] H. Fatta, “*Analisis dan Perancangan Sistem Informasi*”, Yogyakarta: ANDI, 2007.
- [9] G. A. Putra, Endroyono, dan G. Kusrahardjo, “Rancang Bangun *Software* Sistem Monitoring TV Digital DVB-T2”, *Jurnal Teknik ITS*, vol. 4, no. 1, 2015.
- [10] R. A. Pascapraharastyan, A. Supriyanto, dan P. Sudarmaningtyas, “Rancangan Bangun Sistem Informasi Manajemen Arsip Rumah Sakit Bedah Surabaya Berbasis Web”, *Jurnal Sistem Informasi*, vol. 3, no. 1, 2014.

LAMPIRAN I
SEQUENCE DIAGRAM

Lampiran I: Sequence Diagram

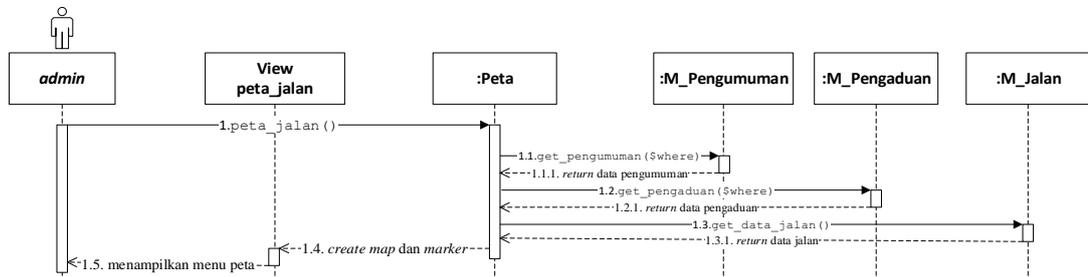
1. Sequence diagram proses login admin



Gambar 1 Sequence diagram proses login admin.

Pada Gambar 1 merupakan *sequence diagram* proses login admin, dimana merupakan proses pertama pada sistem yang digunakan admin. Fungsi `index()` pada class Login dijalankan pertamakali untuk memunculkan form login. Setelah memasukkan `username` dan `password` serta menekan tombol login maka akan memanggil fungsi `cek_login()` pada class Login dan kemudian memanggil fungsi `ceklogin($username)` pada class M_Login untuk memeriksa apakah `username` dan `password` yang dimasukkan ada dalam database. Jika ada atau sesuai maka berarti admin berhasil login ke sistem dan memunculkan pesan sukses, jika data `username` dan `password` tidak ada atau sesuai maka login ke sistem gagal dan memunculkan pesan error.

2. Sequence diagram proses melihat peta oleh admin jalan

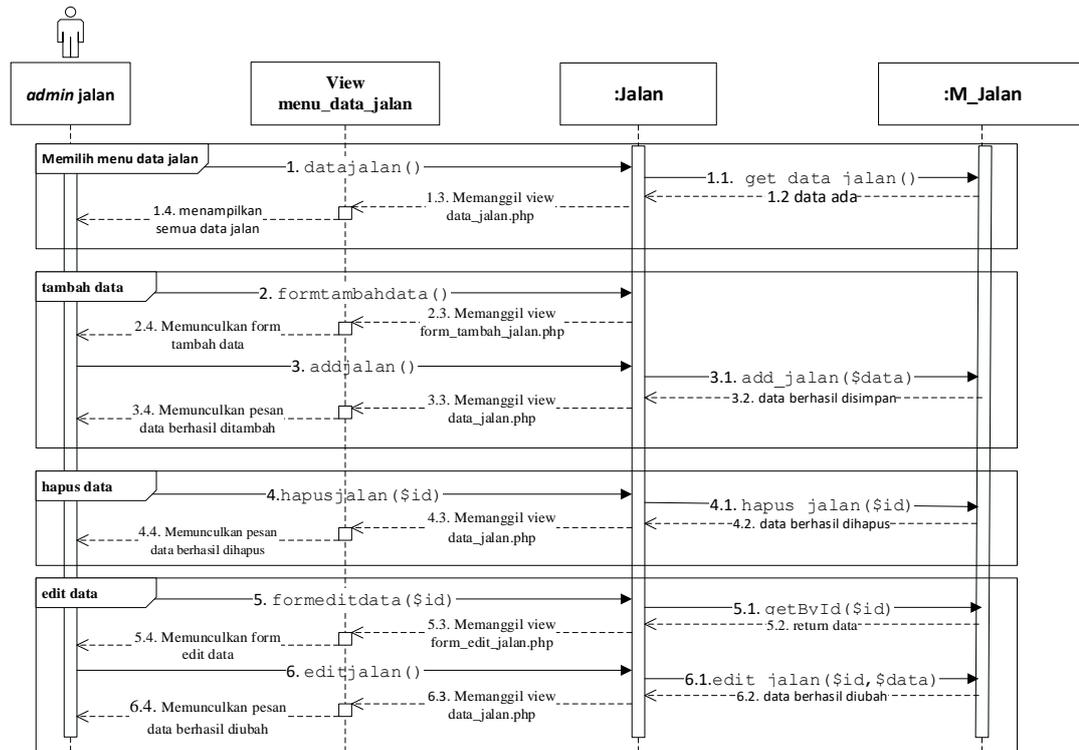


Gambar 2 Sequence diagram proses melihat peta oleh admin jalan.

Pada Gambar 2 merupakan *sequence diagram* proses melihat peta yang digunakan oleh admin jalan. Saat memilih menu peta maka akan menjalankan fungsi `peta_jalan()` pada class Peta dan kemudian memanggil fungsi `get_pengumuman($where)` pada class M_Pengumuman untuk mengambil data

pengumuman tentang jalan, fungsi `get_pengaduan($where)` pada *class* `M_Pengaduan` untuk mengambil data pengaduan tentang jalan, dan fungsi `get_data_jalan()` pada *class* `M_Jalan` untuk mengambil data jalan pada *database* kemudian ditampilkan dalam sebuah peta beserta *marker* dan *polyline* jalan.

3. Sequence diagram proses mengelola data jalan

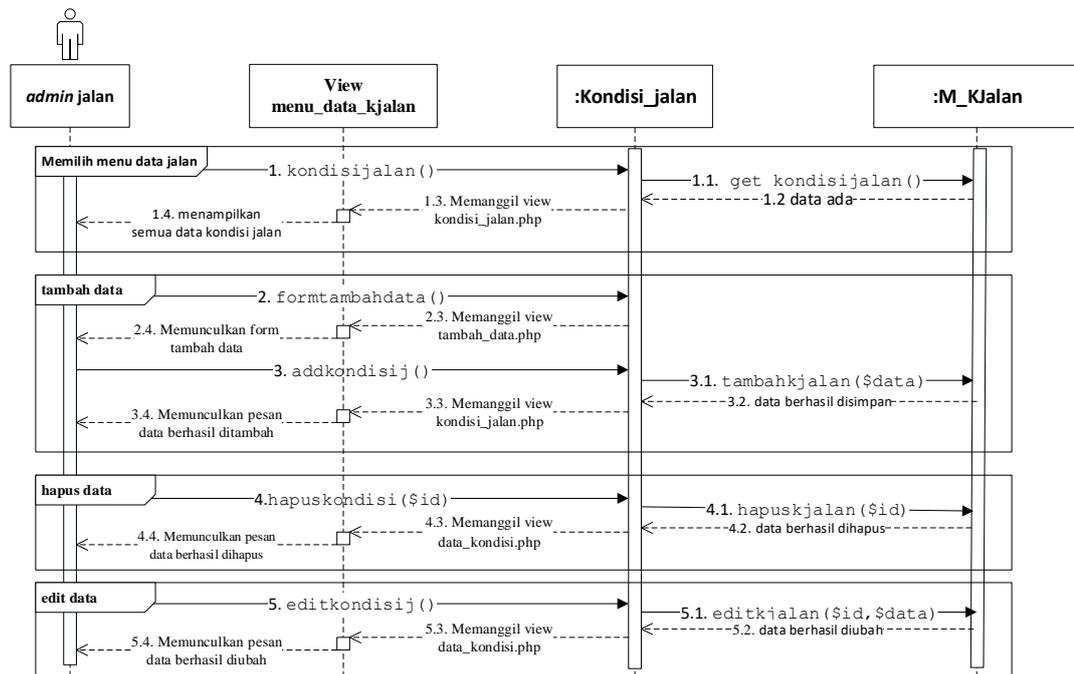


Gambar 3 Sequence diagram proses mengelola data jalan.

Pada Gambar 3 merupakan *sequence* diagram proses mengelola data jalan yang digunakan oleh *admin jalan*. Saat memilih menu data jalan maka akan menjalankan fungsi `data_jalan()` pada *class* `Jalan` dan kemudian memanggil fungsi `get_data_jalan()` pada *class* `M_Jalan` untuk mengambil data jalan pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* `Jalan` untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `addjalan()` pada *class* `Jalan` dan kemudian menjalankan fungsi `add_jalan($data)` pada *class* `M_Jalan` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusjalan($id)` pada *class* `Jalan` kemudian menjalankan fungsi `hapus_jalan($id)` pada *class* `M_Jalan` untuk menghapus data dari *database*. Untuk

edit data, memilih *icon* edit pada salah satu data maka akan memanggil fungsi `formeditdata($id)` pada *class* `Jalan` dan menjalankan fungsi `getById($id)` pada *class* `M_Jalan` untuk mengambil data jalan yang akan diedit pada *database* kemudian menampilkan data tersebut ke halaman *form* edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editjalan()` pada *class* `Jalan` dan kemudian menjalankan fungsi `edit_jalan($id,$data)` pada *class* `M_Jalan` untuk menyimpan perubahan data ke *database*.

4. Sequence diagram proses mengelola data kondisi jalan

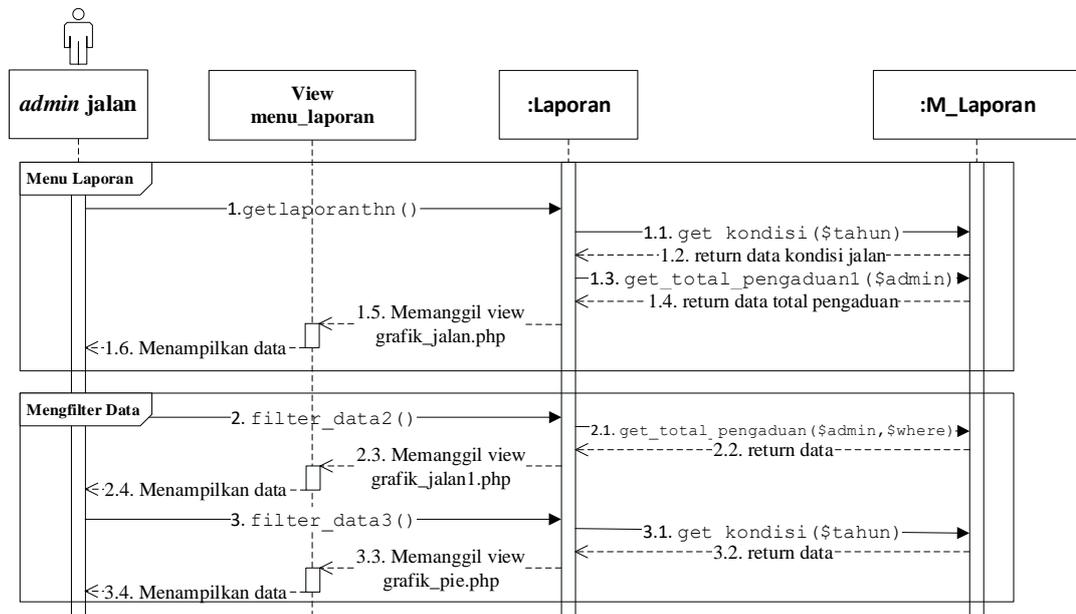


Gambar 4 Sequence diagram proses mengelola data kondisi jalan.

Pada Gambar 4 merupakan *sequence* diagram proses mengelola data kondisi jalan yang digunakan oleh *admin jalan*. Saat memilih menu data kondisi jalan maka akan menjalankan fungsi `kondisijalan()` pada *class* `Kondisi_jalan` dan kemudian memanggil fungsi `get_kondisijalan()` pada *class* `M_KJalan` untuk mengambil data kondisi jalan pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* `Jalan` untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `addkondisij()` pada *class* `Kondisi_jalan` dan kemudian menjalankan fungsi `tambahkjalan($data)` pada *class* `M_KJalan` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil

fungsi `hapusKondisi($id)` pada *class* `Kondisi_jalan` kemudian menjalankan fungsi `hapusKjJalan($id)` pada *class* `M_KJalan` untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan menampilkan modal *form* edit data, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editKondisiJ($id)` pada *class* `Kondisi_jalan` dan kemudian menjalankan fungsi `editKjJalan($id,$data)` pada *class* `M_KJalan` untuk menyimpan perubahan data ke *database*.

5. Sequence diagram proses melihat laporan oleh *admin* jalan

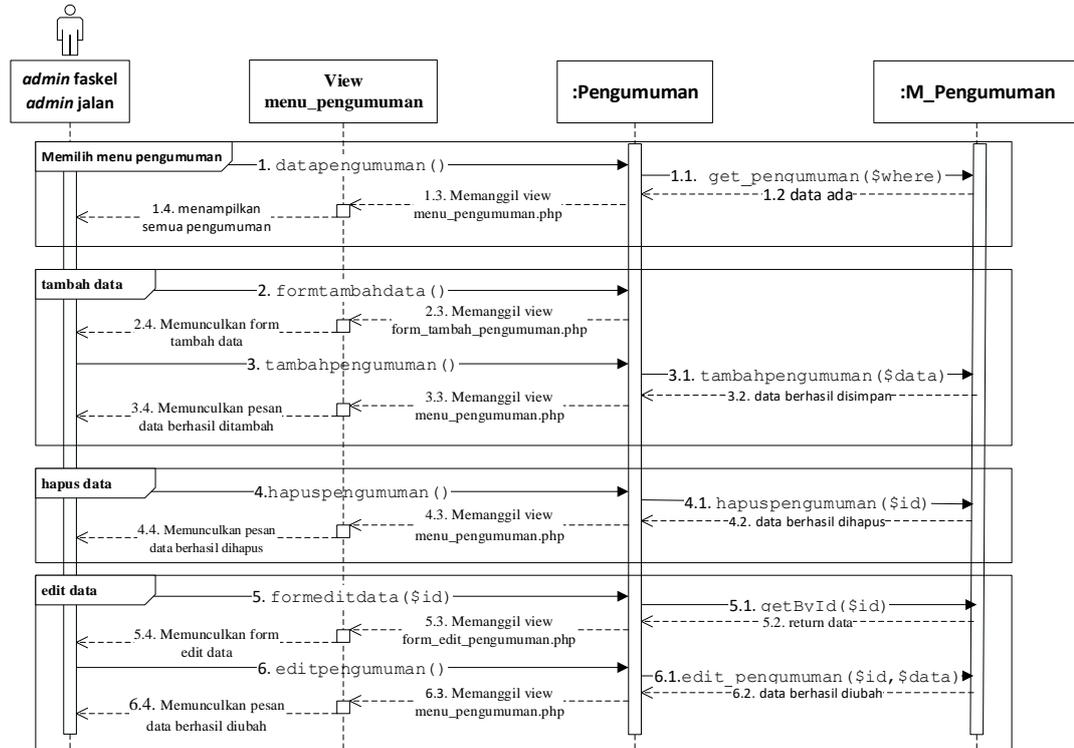


Gambar 5 Sequence diagram proses melihat laporan oleh *admin* jalan.

Pada Gambar 5 merupakan *sequence* diagram proses melihat laporan yang digunakan oleh *admin* jalan. Saat memilih menu laporan maka akan menjalankan fungsi `getlaporanthn()` pada *class* `Laporan` dan kemudian memanggil fungsi `get_kondisi($tahun)` pada *class* `M_Laporan` untuk mengambil data kondisi jalan pada *database* dan fungsi `get_total_pengaduan1($admin)` untuk mengambil data pengaduan jalan pada *database* kemudian ditampilkan dalam bentuk grafik. Terdapat filter data pada menu laporan ini yang dibagi menjadi filter untuk data pengaduan jalan dan filter untuk data kondisinya. Filter untuk data pengaduan berdasarkan rentang tahun, memilih rentang tahun yang diinginkan kemudian menekan tombol filter maka akan menjalankan fungsi `filter_data2()` pada *class* `Laporan` dan kemudian memanggil fungsi `get_total_pengaduan($admin,$where)` pada *class* `M_Laporan` untuk mengambil data dari *database* kemudian dimunculkan dalam bentuk grafik garis. Filter untuk data kondisi jalan berdasarkan tahun tertentu,

memilih tahun yang diinginkan kemudian menekan tombol filter maka akan menjalankan fungsi `filter_data3()` pada *class* Laporan dan kemudian memanggil fungsi `get_kondisi($tahun)` pada *class* M_Laporan untuk mengambil data dari database kemudian dimunculkan dalam bentuk grafik *pie*.

6. Sequence diagram proses mengelola pengumuman

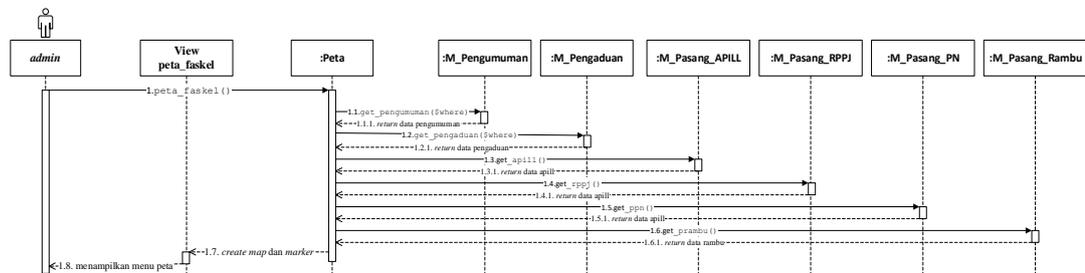


Gambar 6 Sequence diagram proses mengelola pengumuman.

Pada Gambar 6 merupakan *sequence* diagram proses mengelola pengumuman yang digunakan oleh *admin jalan* maupun *admin faskel*. Saat memilih menu pengumuman maka akan menjalankan fungsi `datapengumuman()` pada *class* Pengumuman dan kemudian memanggil fungsi `get_pengumuman($where)` pada *class* M_Pengumuman untuk mengambil data pengumuman pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* Pengumuman untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `tambahpengumuman()` pada *class* Pengumuman. dan kemudian menjalankan fungsi `tambahpengumuman($data)` pada *class* M_Pengumuman untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi

hapuspengumuman() pada class Pengumuman kemudian menjalankan fungsi hapuspengumuman(\$id) pada class M_Pengumuman untuk menghapus data dari database. Untuk edit data, memilih icon edit pada salah satu data maka akan memanggil fungsi formeditdata(\$id) pada class Pengumuman dan menjalankan fungsi getById(\$id) pada class M_Pengumuman untuk mengambil data jalan yang akan diedit pada database kemudian menampilkan data tersebut ke halaman form edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi editpengumuman() pada class Pengumuman dan kemudian menjalankan fungsi edit_pengumuman(\$id,\$data) pada class M_Pengumuman untuk menyimpan perubahan data ke database.

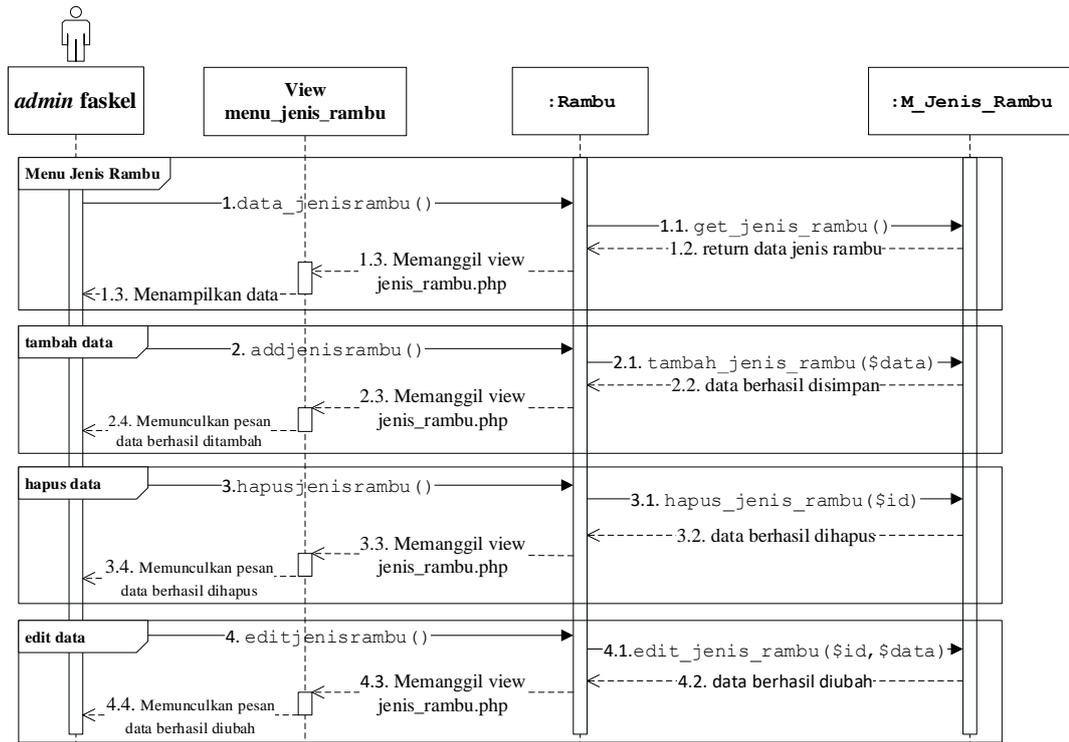
7. Sequence diagram proses melihat peta oleh admin faskel



Gambar 7 Sequence diagram proses melihat peta oleh admin faskel.

Pada Gambar 7 merupakan sequence diagram proses melihat peta yang digunakan oleh admin faskel. Saat memilih menu peta maka akan menjalankan fungsi peta_faskel() pada class Peta dan kemudian memanggil fungsi get_pengumuman(\$where) pada class M_Pengumuman untuk mengambil data pengumuman tentang jalan, fungsi get_pengaduan(\$where) pada class M_Pengaduan untuk mengambil data pengaduan tentang jalan, fungsi get_apill() pada class M_Pasang_APILL untuk mengambil data pemasangan APILL, fungsi get_rppj() pada class M_Pasang_RPPJ untuk mengambil data pemasangan RPPJ, fungsi get_ppn() pada class M_Pasang_PN untuk mengambil data pemasangan papan nama jalan, dan fungsi get_prambu() pada class M_Pasang_Rambu untuk mengambil data pemasangan rambu pada database kemudian ditampilkan dalam sebuah peta beserta marker-nya.

8. Sequence diagram proses mengelola jenis rambu

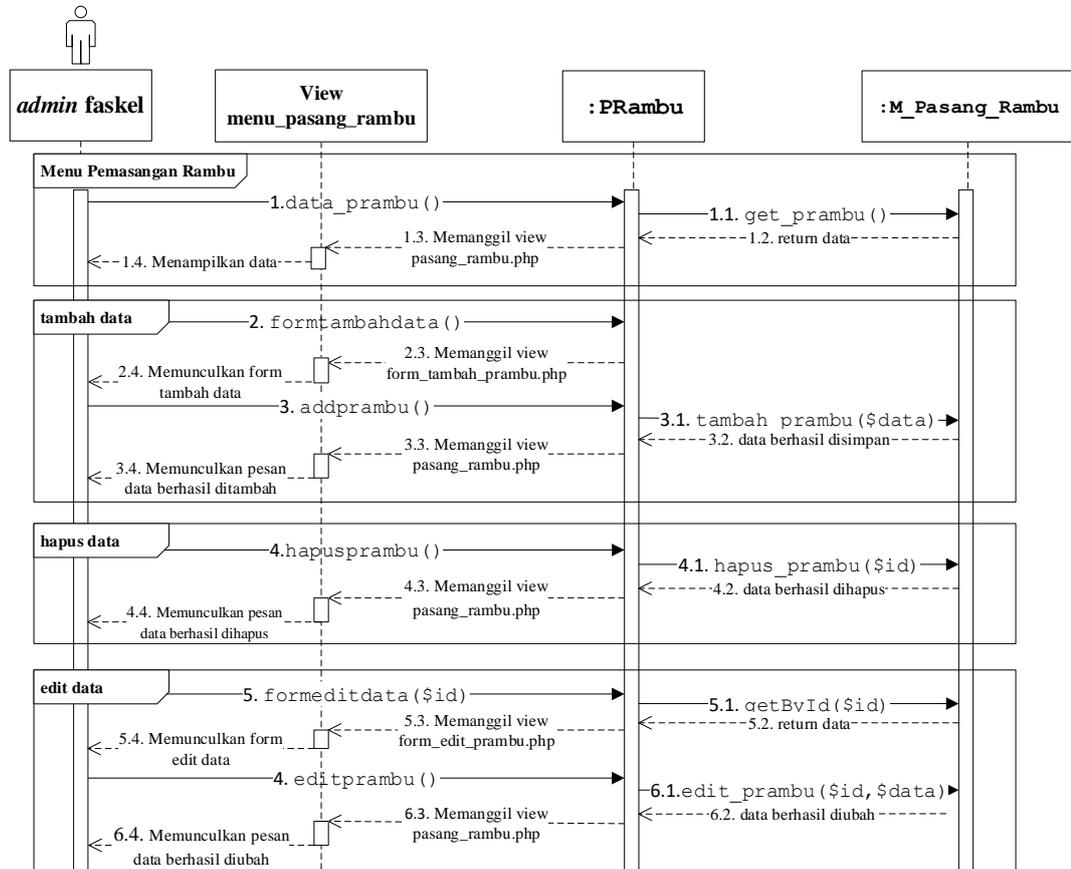


Gambar 8 Sequence diagram proses mengelola jenis rambu.

Pada Gambar 8 merupakan *sequence* diagram proses mengelola jenis rambu yang digunakan oleh *admin faskel*. Saat memilih menu data jenis rambu maka akan menjalankan fungsi `data_jenisrambu()` pada *class* `Rambu` dan kemudian memanggil fungsi `get_jenis_rambu()` pada *class* `M_Jenis_Rambu` untuk mengambil data jenis rambu pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil modal *form* tambah data, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `addjenisrambu()` pada *class* `Rambu` dan kemudian menjalankan fungsi `tambah_jenis_rambu($data)` pada *class* `M_Jenis_Rambu` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusjenisrambu()` pada *class* `Rambu` kemudian menjalankan fungsi `hapus_jenis_rambu($id)` pada *class* `M_Jenis_Rambu` untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan menampilkan modal *form* edit data, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editjenisrambu()` pada *class* `Rambu` dan

kemudian menjalankan fungsi `edit_jenis_rambu($id,$data)` pada *class* `M_Jenis_Rambu` untuk menyimpan perubahan data ke *database*.

9. Sequence diagram proses mengelola data pemasangan rambu

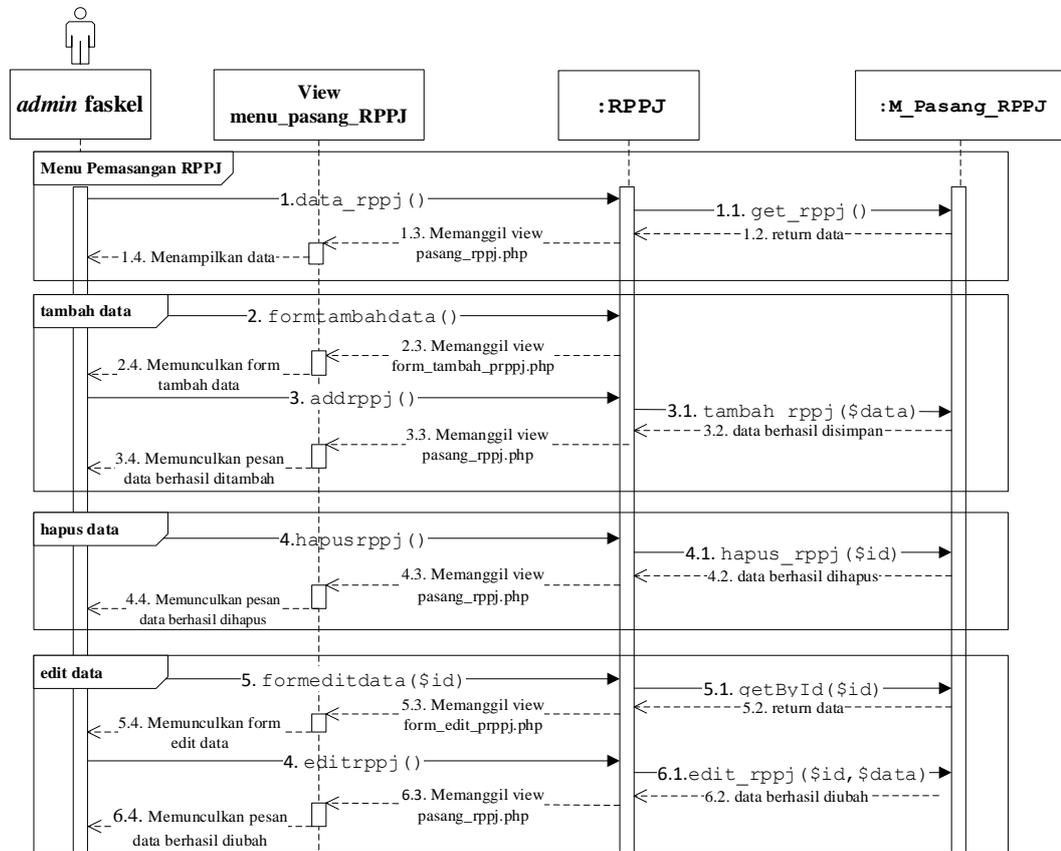


Gambar 9 Sequence diagram proses mengelola data pemasangan rambu.

Pada Gambar 9 merupakan *sequence diagram* proses mengelola data pemasangan rambu yang digunakan oleh *admin faskel*. Saat memilih menu data pemasangan rambu maka akan menjalankan fungsi `data_prambu()` pada *class* `PRambu` dan kemudian memanggil fungsi `get_prambu()` pada *class* `M_Pasang_Rambu` untuk mengambil data pemasangan rambu pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* `PRambu` untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `addprambu()` pada *class* `PRambu` dan kemudian menjalankan fungsi `tambah_prambu($data)` pada *class* `M_Pasanga_Rambu` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusprambu()` pada *class* `PRambu` kemudian menjalankan

fungsi `hapus_prambu($id)` pada *class* `M_Pemasangan_Rambu` untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan memanggil fungsi `formeditdata($id)` pada *class* `PRambu` dan menjalankan fungsi `getById($id)` pada *class* `M_Pemasangan_Rambu` untuk mengambil data jalan yang akan diedit pada *database* kemudian menampilkan data tersebut ke halaman *form* edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editprambu()` pada *class* `PRambu` dan kemudian menjalankan fungsi `edit_prambu($id,$data)` pada *class* `M_Pemasangan_Rambu` untuk menyimpan perubahan data ke *database*.

10. Sequence diagram proses mengelola data pemasangan RPPJ

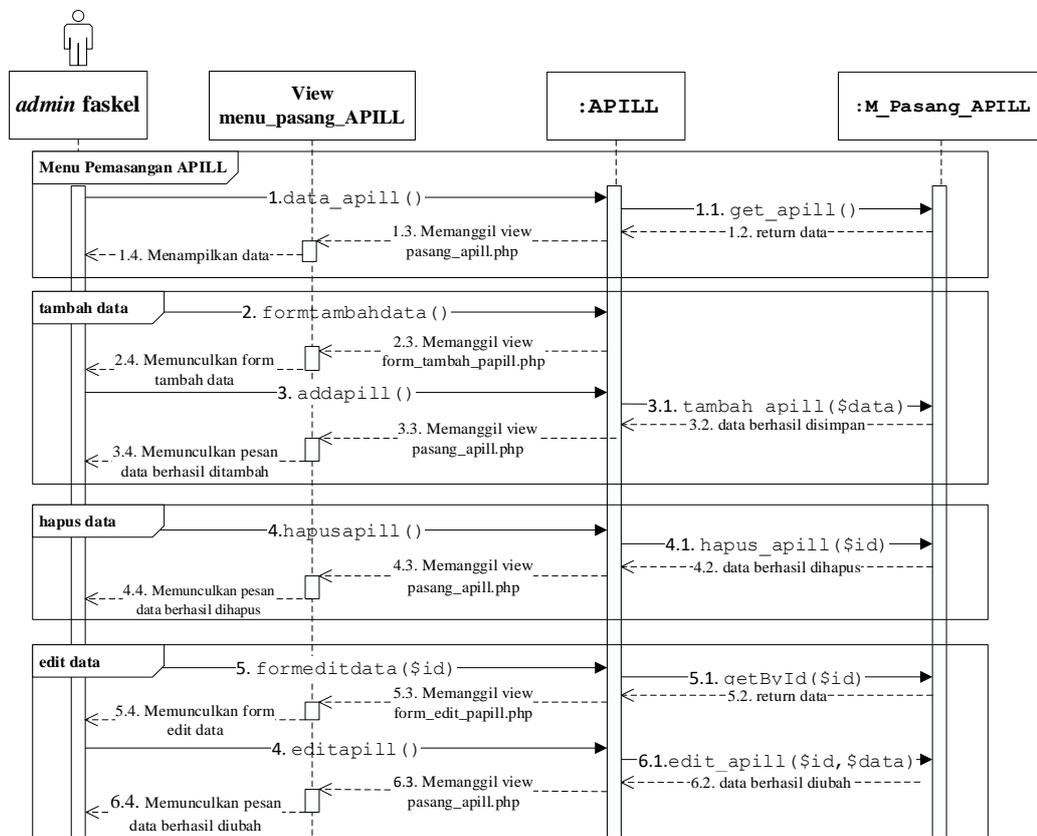


Gambar 10 Sequence diagram proses mengelola data pemasangan RPPJ.

Pada Gambar 10 merupakan *sequence* diagram proses mengelola data pemasangan RPPJ yang digunakan oleh *admin faskel*. Saat memilih menu data pemasangan RPPJ maka akan menjalankan fungsi `data_rppj()` pada *class* `RPPJ` dan kemudian memanggil fungsi `get_rppj()` pada *class* `M_Pasang_RPPJ` untuk mengambil data pemasangan rppj pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data.

Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* RPPJ untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan, pada tombol simpan akan memanggil fungsi `addrppj()` pada *class* RPPJ dan kemudian menjalankan fungsi `tambah_rppj($data)` pada *class* M_Pasanga_RPPJ untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusrppj()` pada *class* RPPJ kemudian menjalankan fungsi `hapus_rppj($id)` pada *class* M_Pemasangan_RPPJ untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan memanggil fungsi `formeditdata($id)` pada *class* RPPJ dan menjalankan fungsi `getById($id)` pada *class* M_Pemasangan_RPPJ untuk mengambil data RPPJ yang akan diedit pada *database* kemudian menampilkan data tersebut ke halaman *form* edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editrppj()` pada *class* RPPJ dan kemudian menjalankan fungsi `edit_rppj($id,$data)` pada *class* M_Pemasangan_RPPJ untuk menyimpan perubahan data ke *database*.

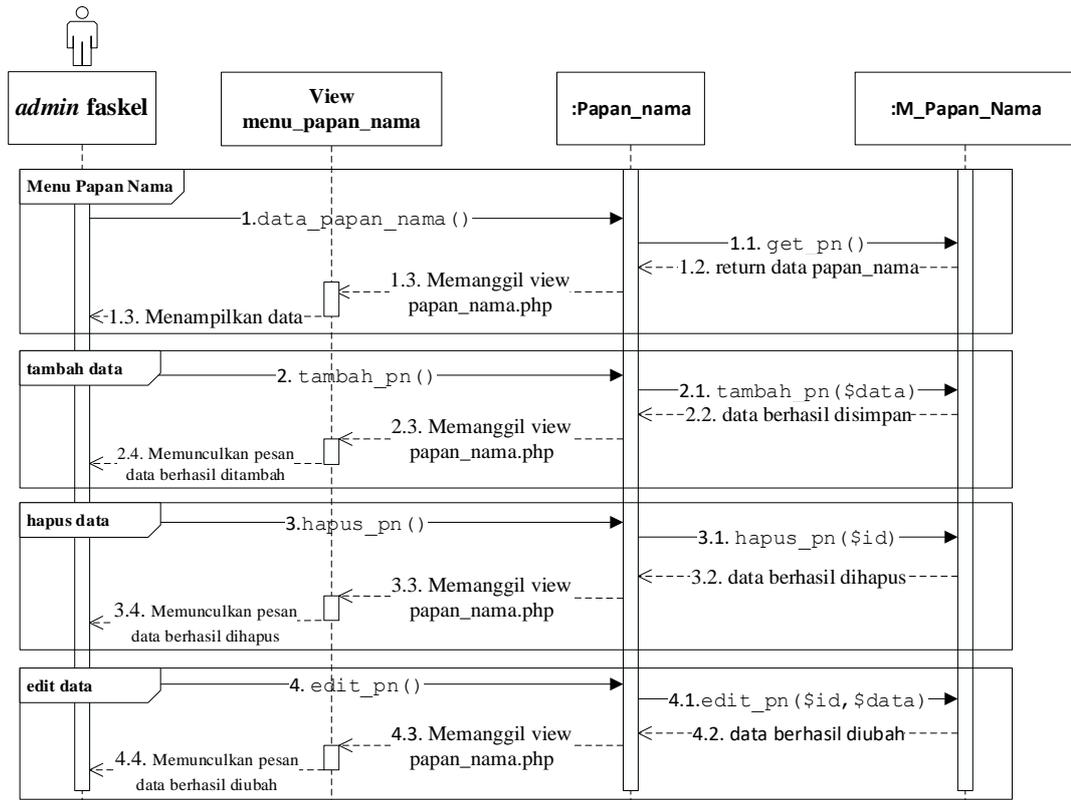
11. Sequence diagram proses mengelola data pemasangan APILL



Gambar 11 Sequence diagram proses mengelola data pemasangan APILL.

Pada Gambar 11 merupakan *sequence* diagram proses mengelola data pemasangan APILL yang digunakan oleh *admin* faskel. Saat memilih menu data pemasangan APILL maka akan menjalankan fungsi `data_apill()` pada *class* APILL dan kemudian memanggil fungsi `get_apill()` pada *class* M_Pasang_APILL untuk mengambil data pemasangan apill pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* APILL untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan, pada tombol simpan akan memanggil fungsi `addapill()` pada *class* APILL dan kemudian menjalankan fungsi `tambah_apill($data)` pada *class* M_Pasanga_APILL untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusapill()` pada *class* APILL kemudian menjalankan fungsi `hapus_apill($id)` pada *class* M_Pemasangan_APILL untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan memanggil fungsi `formeditdata($id)` pada *class* APILL dan menjalankan fungsi `getById($id)` pada *class* M_Pemasangan_APILL untuk mengambil data APILL yang akan diedit pada *database* kemudian menampilkan data tersebut ke halaman *form* edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editapill()` pada *class* APILL dan kemudian menjalankan fungsi `edit_apill($id,$data)` pada *class* M_Pemasangan_APILL untuk menyimpan perubahan data ke *database*.

12. Sequence diagram proses mengelola data papan nama

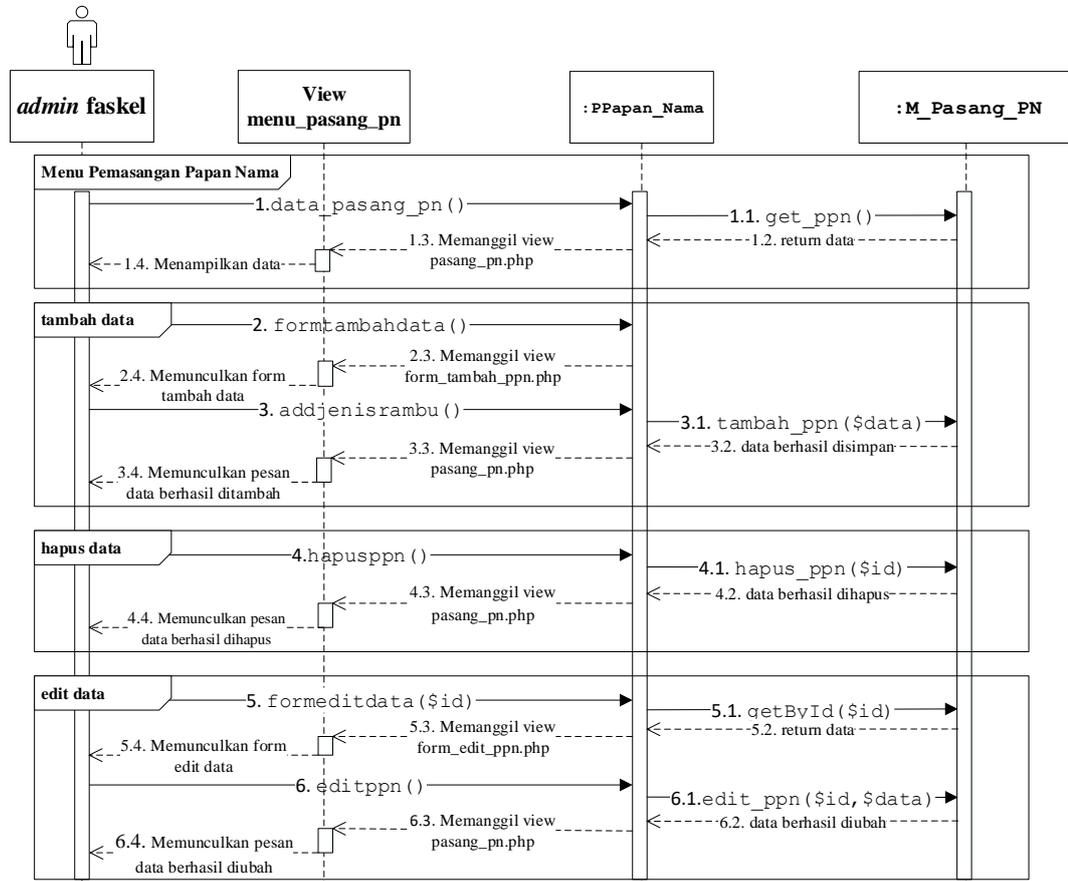


Gambar 12 Sequence diagram proses mengelola data papan nama.

Pada Gambar 12 merupakan *sequence* diagram proses mengelola data papan nama yang digunakan oleh *admin faskel*. Saat memilih menu data jenis rambu maka akan menjalankan fungsi `data_papan_nama()` pada *class* `Papan_nama` dan kemudian memanggil fungsi `get_pn()` pada *class* `M_Papan_Nama` untuk mengambil data papan nama pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil modal *form* tambah data, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `tambah_pn()` pada *class* `Papan_nama` dan kemudian menjalankan fungsi `tambah_pn($data)` pada *class* `M_Papan_Nama` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapus_pn()` pada *class* `Papan_nama` kemudian menjalankan fungsi `hapus_pn($id)` pada *class* `M_Papan_Nama` untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan menampilkan modal *form* edit data, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `edit_pn()` pada *class* `Papan_nama` dan kemudian menjalankan

fungsi `edit_pn($id,$data)` pada *class* `M_Papan_Nama` untuk menyimpan perubahan data ke *database*.

13. Sequence diagram proses mengelola data pemasangan papan nama

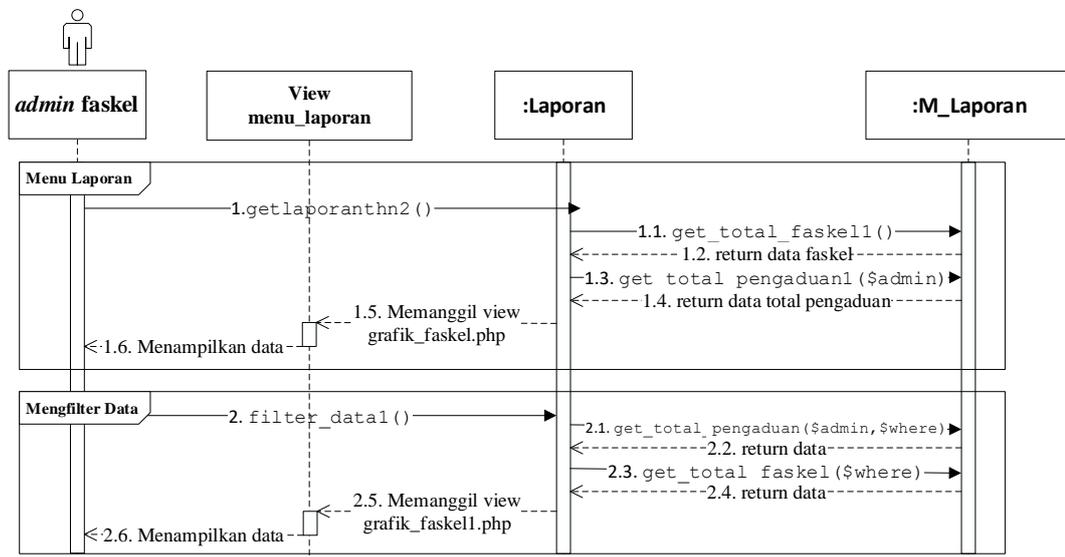


Gambar 13 Sequence diagram proses mengelola data pemasangan papan nama.

Pada Gambar 13 merupakan *sequence* diagram proses mengelola data pemasangan papan nama yang digunakan oleh *admin faskel*. Saat memilih menu data pemasangan papan nama jalan maka akan menjalankan fungsi `data_pasang_pn()` pada *class* `PPapan_Nama` dan kemudian memanggil fungsi `get_ppn()` pada *class* `M_Pasang_PN` untuk mengambil data pemasangan papan nama jalan pada *database* kemudian ditampilkan dalam bentuk tabel. Ada tiga proses utama yang dapat dilakukan yaitu tambah data, hapus data, dan edit data. Untuk tambah data maka memilih tombol tambah data akan memanggil fungsi `formtambahdata()` pada *class* `PPapan_Nama` untuk mengarahkan ke halaman *form*, kemudian untuk proses simpan pada tombol simpan akan memanggil fungsi `addppn()` pada *class* `PPapan_Nama` dan kemudian menjalankan fungsi `tambah_ppn($data)` pada *class* `M_Pasanga_PN` untuk menyimpan data ke *database*. Untuk hapus data maka memilih *icon* hapus pada salah satu data akan memanggil fungsi `hapusppn()` pada *class* `PPapan_Nama`

kemudian menjalankan fungsi `hapus_ppn($id)` pada *class* `M_Pemasangan_PN` untuk menghapus data dari *database*. Untuk edit data, memilih *icon* edit pada salah satu data maka akan memanggil fungsi `formeditdata($id)` pada *class* `PPapan_Nama` dan menjalankan fungsi `getById($id)` pada *class* `M_Pemasangan_PN` untuk mengambil data pemasangan papan nama yang akan diedit pada *database* kemudian menampilkan data tersebut ke halaman *form* edit, kemudian untuk proses edit pada tombol simpan akan memanggil fungsi `editppn()` pada *class* `PPapan_Nama` dan kemudian menjalankan fungsi `edit_ppn($id, $data)` pada *class* `M_Pemasangan_PN` untuk menyimpan perubahan data ke *database*.

14. Sequence diagram proses melihat laporan oleh admin faskel

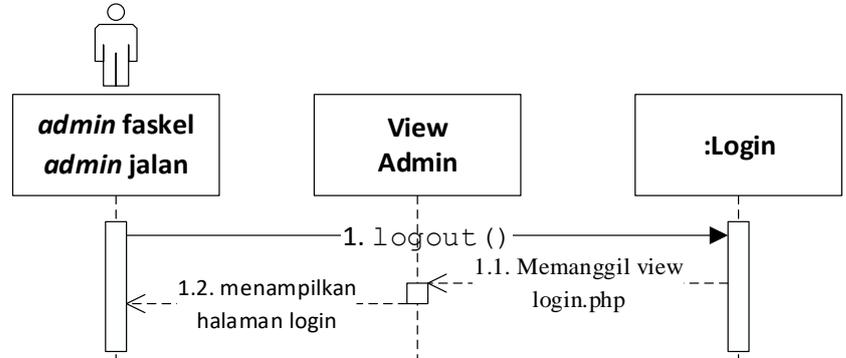


Gambar 14 Sequence diagram proses melihat laporan oleh admin faskel.

Pada Gambar 14 merupakan *sequence* diagram proses melihat laporan yang digunakan oleh *admin* faskel. Saat memilih menu laporan maka akan menjalankan fungsi `getlaporanthn2()` pada *class* `Laporan` dan kemudian memanggil fungsi `get_total_faskel1()` pada *class* `M_Laporan` untuk mengambil data semua fasilitas kelengkapan pada *database* dan fungsi `get_total_pengaduan1($admin)` untuk mengambil data pengaduan jalan pada *database* kemudian ditampilkan dalam bentuk grafik. Terdapat filter data pengaduan jalan dan filter data fasilitas kelengkapan jalan berdasarkan rentang tahun. Mulai dari memilih rentang tahun yang diinginkan kemudian menekan tombol filter maka akan menjalankan fungsi `filter_data1()` pada *class* `Laporan`, kemudian memanggil fungsi `get_total_pengaduan($admin, $where)` pada *class* `M_Laporan` dan fungsi

`get_total_faskel($where)` pada *class* `M_Laporan` untuk mengambil data dari database kemudian dimunculkan dalam bentuk dua grafik garis.

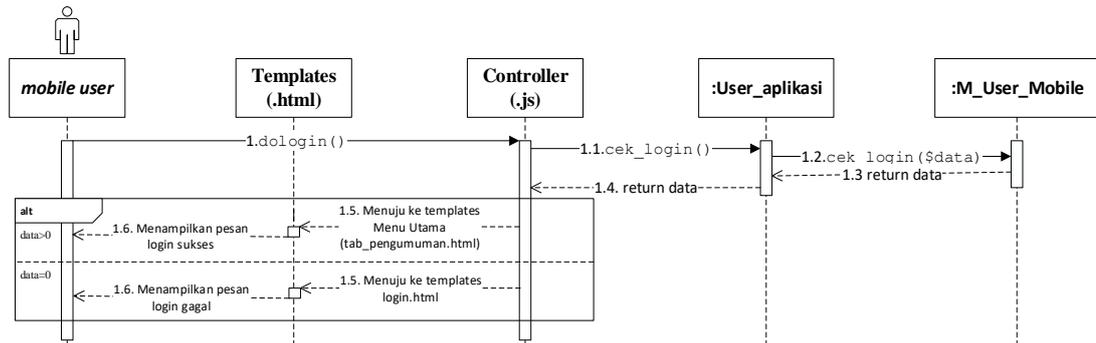
15. Sequence diagram proses *logout* aplikasi web



Gambar 15 Sequence diagram proses *logout* aplikasi web.

Pada Gambar 15 merupakan *sequence* diagram proses *logout* yang digunakan oleh *admin faskel* dan *admin jalan*. Saat memilih menu *logout* maka akan menjalankan fungsi `logout()` pada *class* `Login` dan mengarahkan ke halaman *login*.

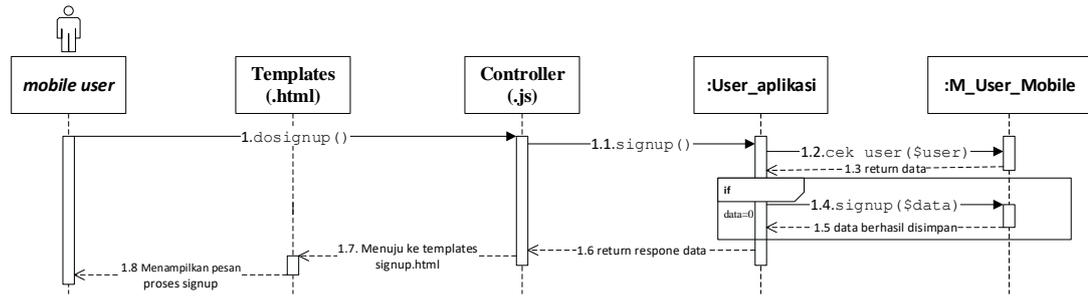
16. Sequence diagram proses *login* aplikasi mobile



Gambar 16 Sequence diagram proses *login* aplikasi mobile.

Pada Gambar 16 merupakan *sequence* diagram proses *login* aplikasi *mobile* yang digunakan oleh masyarakat. Saat aplikasi *mobile* dijalankan yang akan tampil pertamakali adalah *form login*. Actor pengguna aplikasi jika sudah membuat akun tinggal mengisi *username* dan *password* akunya, kemudian menekan tombol *login* yang akan menjalankan fungsi `dologin()` pada *class* `loginCtrl.js`. Dalam fungsi tersebut akan melakukan *request* pengecekan data ke *server* dengan memanggil fungsi `cek_login()` pada *class* `User_aplikasi`, dalam fungsi *class* *controller server* ini akan memanggil fungsi `cek_login($data)` pada *class* `M_User_Mobile` untuk mengecek keberadaan data dalam *database*. Jika data sesuai dengan data yang ada di *database server* maka sistem akan mengarahkan ke halaman *home*, jika data tidak sesuai maka akan menampilkan pesan *error* pada halaman *login* tersebut.

17. Sequence diagram proses *signup* aplikasi *mobile*



Gambar 17 Sequence diagram proses *signup* aplikasi *mobile*.

Pada Gambar 17 merupakan *sequence* diagram proses *signup* aplikasi *mobile* yang digunakan oleh *mobile user*. Melakukan pembuatan akun untuk login ke sistem aplikasi *mobile* hanya perlu mengisi form signup kemudian meng-submit data, kemudian sistem akan menjalankan fungsi `dosignup()` pada *class* `signupCtrl`. Data akan dikirim ke server dengan memanggil fungsi `signup()` pada *class* `User_aplikasi` dan akan menjalankan fungsi `cek_user($user)` pada *class* `M_User_Mobile` yang bertujuan untuk melihat apakah *username* yang digunakan sudah ada pada *database*, hal ini ditujukan agar tidak terdapat *username* yang sama. Jika *username* tersebut belum ada dalam *database* sistem maka akan menjalankan fungsi `signup($data)` pada *class* `M_User_Mobile` untuk menyimpan seluruh data akun baru tersebut ke *database*.

LAMPIRAN II
IMPLEMENTASI TABEL PADA DATABASE

Lampiran II: Implementasi Tabel Pada Database

Pada *database* sistem yang dibuat terdiri dari 14 tabel yaitu table admin, tabel feedback, tabel jalan, tabel jenis_rambu, tabel kondisi_jalan, tabel melakukan_pengaduan (tabel relasi antar tabel pengaduan dan tabel mobile_user), tabel mobile_user, tabel papan_nama, tabel pasang_apill, tabel pasang_pn, tabel pasang_rambu, tabel pasang_rppj, tabel pengaduan, dan tabel pengumuman. Berikut merupakan penjelasan dari setiap tabel yang ada.

1. Tabel admin

Tabel admin berisi data pengguna aplikasi *web* atau admin yang merupakan perwakilan dari Dinas Perhubungan Kota Mataram dan juga dari Dinas Pekerjaan Umum dan Tatanan Ruang Kota Mataram. *Primary key* dari tabel admin adalah *field* idAdmin dengan tipe data char(3). Rincian *field-field* dalam tabel admin dapat dilihat pada Gambar 18.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	idAdmin	char(3)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	namaAdmin	varchar(15)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	passAdmin	varchar(15)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	hak_akses	enum('jalan kota', 'faskel kota')			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 18 Tabel admin.

2. Tabel mobile_user

Tabel mobile_user berisi data pengguna aplikasi *mobile* atau masyarakat Kota Mataram. *Primary key* dari tabel mobile_user adalah *field* id_user dengan tipe data varchar(10). Rincian *field-field* dalam tabel mobile_user dapat dilihat pada Gambar 19.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_user	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	nama_user	varchar(30)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	alamat_user	varchar(50)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	nohp_user	varchar(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	email_user	varchar(15)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
6	passwordU	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 19 Tabel mobile_user.

3. Tabel jalan

Tabel jalan berisi data tentang jalan di Kota Mataram. *Primary key* dari tabel jalan adalah *field* kode_jalan dengan tipe data varchar(12). Rincian *field-field* dalam tabel jalan dapat dilihat pada Gambar 20.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_jalan	varchar(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	nama_jalan	varchar(30)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	jenis_permukaan	varchar(20)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	panjangRuas	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	lebarRuas	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
6	kecamatan	varchar(20)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
7	kelurahan	varchar(30)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
8	status_jalan	enum('Jalan Kota', 'Jalan Nasional', 'Jalan Provin...			No	None		Change Drop Primary Unique Index Spatial Fulltext More
9	koordinat	text			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 20 Tabel jalan.

4. Tabel kondisi_jalan

Tabel kondisi_jalan berisi data kondisi jalan Kota Mataram. *Primary key* dari tabel kondisi_jalan adalah *field* idKondisi dengan tipe data varchar(10). Rincian *field-field* dalam tabel kondisi_jalan dapat dilihat pada Gambar 21.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	idKondisi	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	kode_jalan	varchar(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	kBaik	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	kSedang	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	kRusakRingan	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
6	kRusakBerat	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
7	tahun_data	year(4)			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 21 Tabel kondisi_jalan.

5. Tabel jenis_rambu

Tabel jenis_rambu berisi data jenis rambu-rambu lalu lintas. *Primary key* dari tabel jenis_rambu adalah *field* kode_rambu dengan tipe data varchar(12). Rincian *field-field* dalam tabel jenis_rambu dapat dilihat pada Gambar 22.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_rambu	varchar(12)			No	None		Change Drop Primary Unique Index More
2	kategori_rambu	enum('Larangan', 'Petunjuk', 'Peringatan', 'Perint...			No	None		Change Drop Primary Unique Index More
3	keterangan_rambu	text			No	None		Change Drop Primary Unique Index More
4	gambar_rambu	varchar(20)			No	default.jpg		Change Drop Primary Unique Index More

Gambar 22 Tabel jenis_rambu.

6. Tabel pasang_rambu

Tabel pasang_rambu berisi data pemasangan rambu-rambu lalu lintas. *Primary key* dari tabel pasang_rambu adalah *field* id_pasang_rambu dengan tipe data varchar(15). Rincian *field-field* dalam tabel pasang_rambu dapat dilihat pada Gambar 23.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_pasang_rambu	varchar(15)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
2	kode_rambu	varchar(12)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
3	kode_jalan	varchar(12)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
4	lat_rambu	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
5	lng_rambu	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
6	ket_lokasi	varchar(30)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
7	tahun_data	year(4)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
8	gambar	varchar(20)			No	default.jpg	Change Drop Primary Unique Index Spatial Fulltext	More

Gambar 23 Tabel pasang_rambu.

7. Tabel papan_nama

Tabel papan_nama berisi data papan nama jalan. *Primary key* dari tabel papan_nama adalah *field* id_pn dengan tipe data varchar(15). Rincian *field-field* dalam tabel papan_nama dapat dilihat pada Gambar 24.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_pn	varchar(15)			No	None	Change Drop Primary Unique Index Spatial Fulltext Distinct values	More
2	kode_jalan	varchar(12)			No	None	Change Drop Primary Unique Index Spatial Fulltext Distinct values	More
3	gambar	varchar(15)			No	default.jpg	Change Drop Primary Unique Index Spatial Fulltext Distinct values	More

Gambar 24 Tabel papan_nama.

8. Tabel pasang_pn

Tabel pasang_pn berisi data pemasangan papan nama jalan di Kota Mataram. *Primary key* dari tabel pasang_pn adalah *field* id_pasang_pn dengan tipe data varchar(15). Rincian *field-field* dalam tabel pasang_pn dapat dilihat pada Gambar 25.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_pasang_pn	varchar(15)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
2	id_pn	varchar(15)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
3	lat_pn	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
4	lng_pn	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
5	ket_lokasi	varchar(30)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
6	tahun_data	year(4)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
7	gambar	varchar(20)			No	default.jpg	Change Drop Primary Unique Index Spatial Fulltext	More

Gambar 25 Tabel pasang_pn.

9. Tabel pasang_apill

Tabel pasang_apill berisi data pemasangan APILL di Kota Mataram. *Primary key* dari tabel pasang_apill adalah *field* id_apill dengan tipe data varchar(15). Rincian *field-field* dalam tabel pasang_apill dapat dilihat pada Gambar 26.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_apill	varchar(15)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
2	kode_jalan	varchar(12)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
3	lat_apill	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
4	lng_apill	double			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
5	ket_lokasi	varchar(30)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
6	tahun_data	year(4)			No	None	Change Drop Primary Unique Index Spatial Fulltext	More
7	gambar	varchar(20)			No	default.jpg	Change Drop Primary Unique Index Spatial Fulltext	More

Gambar 26 Tabel pasang_apill.

10. Tabel pasang_rppj

Tabel pasang_rppj berisi data pemasangan RPPJ di Kota Mataram. *Primary key* dari tabel pasang_rppj adalah *field* id_rppj dengan tipe data varchar(15). Rincian *field-field* dalam tabel pasang_rppj dapat dilihat pada Gambar 27.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_rppj	varchar(15)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	kode_jalan	varchar(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	lat_rppj	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	lng_rppj	double			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	ket_lokasi	varchar(30)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
6	tahun_data	year(4)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
7	gambar	varchar(20)			No	default.jpg		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 27 Tabel pasang_rppj.

11. Tabel pengumuman

Tabel pengumuman berisi data pengumuman jika dilakukan pembangunan dan perbaikan jalan maupun fasilitas kelengkapan jalan di Kota Mataram. *Primary key* dari tabel pengumuman adalah *field* kode dengan tipe data varchar(15). Rincian *field-field* dalam tabel pengumuman dapat dilihat pada Gambar 28.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode	varchar(15)			No	None		Change Drop Primary Unique Index Spatial More
2	idAdmin	varchar(10)			No	None		Change Drop Primary Unique Index Spatial More
3	judulP	varchar(50)			No	None		Change Drop Primary Unique Index Spatial More
4	kategori	varchar(10)			No	None		Change Drop Primary Unique Index Spatial More
5	isiPengumuman	text			No	None		Change Drop Primary Unique Index Spatial More
6	waktu_upload	datetime			No	None		Change Drop Primary Unique Index Spatial More
7	status	enum('Aktif', 'Sembunyikan')			No	None		Change Drop Primary Unique Index Spatial More
8	fotoP	varchar(20)			No	default.jpg		Change Drop Primary Unique Index Spatial More
9	latitudP	double			No	None		Change Drop Primary Unique Index Spatial More
10	longitudP	double			No	None		Change Drop Primary Unique Index Spatial More

Gambar 28 Tabel pengumuman.

12. Tabel pengaduan

Tabel pengaduan berisi data pengaduan yang dilakukan oleh masyarakat pengguna aplikasi *mobile*. *Primary key* dari tabel pengaduan adalah *field* idPengaduan dengan tipe data varchar(10). Rincian *field-field* dalam tabel pengaduan dapat dilihat pada Gambar 29.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	idPengaduan	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
2	kategoriPengaduan	varchar(25)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
3	judul_pengaduan	varchar(25)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
4	deskripsi	text			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
5	status	enum('Belum dibaca', 'Tampilkan', 'Sembunyikan')			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
6	latitud	double			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
7	longitud	double			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
8	foto	varchar(25)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns

Gambar 29 Tabel pengaduan.

13. Tabel melakukan_pengaduan

Tabel melakukan_pengaduan merupakan tabel relasi antara tabel pengaduan dan tabel mobile_user, berisi tentang keterangan user yang melakukan pengaduan. *Primary key* dari tabel melakukan_pengaduan adalah *field* id dengan tipe data varchar(10). Rincian *field-field* dalam tabel melakukan pengaduan dapat dilihat pada Gambar 30.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	iduser	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	idPengaduan	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	waktu	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 30 Tabel melakukan_pengaduan.

14. Tabel feedback

Tabel *feedback* berisi data tanggapan pengaduan oleh admin. *Primary key* dari tabel *feedback* adalah *field* idFeedback dengan tipe data varchar(15). Rincian *field-field* dalam tabel *feedback* dapat dilihat pada Gambar 31.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	idFeedback	varchar(15)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	idPenanggap	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
3	idPengaduan	varchar(10)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	isi_tanggapan	text			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	waktu_tanggapan	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext More

Gambar 31 Tabel *feedback*.

LAMPIRAN III
HASIL PENGUJIAN METODE BLACK BOX

Lampiran III: Hasil Pengujian Metode Black Box

1. Pengujian aplikasi untuk *admin*

Pengujian aplikasi web untuk admin menggunakan metode *black box* di lakukan pada beberapa fitur yaitu login, mengelola data jalan, mengelola kondisi jalan, mengelola data papan nama jalan, mengelola data pemasangan papan nama jalan, mengelola data jenis rambu, mengelola data pemasangan rambu, mengelola data pemasangan apill, mengelola data pemasangan rppj, mengelola pengumuman, mengelola data pengaduan, memberikan tanggapan pengaduan, dan melihat grafik tahunan.

c. Pengujian fitur *login* aplikasi *web*

Pengujian fitur *login* dilakukan untuk menguji apakah fungsi yang berjalan pada halaman *login* sudah berjalan dengan baik dan benar. Pengujian fitur *login* akan dijelaskan pada Tabel 1.

Tabel 1 Pengujian fitur *login* aplikasi *web*.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
<i>Form username</i> dan <i>password</i> di isi dengan data yang tidak sesuai.	Proses <i>login</i> tidak berhasil dan memunculkan pesan <i>error</i> .	Sesuai	Valid
<i>Form username</i> dan <i>password</i> di isi dengan data yang sesuai.	Proses <i>login</i> berhasil dan di arahkan ke halaman menu Peta.	Sesuai	Valid

Pada pengujian Tabel 1 terdapat beberapa varian dalam input data yaitu apabila form login diisi dengan data tidak sesuai, maka login tidak berhasil dan akan memunculkan pesan error. Sedangkan bila form login diisi dengan data sesuai dengan data yang tersimpan pada database, maka admin berhasil login dan diarahkan ke halaman menu Peta.

d. Pengujian mengelola data jalan

Pengujian mengelola data jalan dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data jalan sudah berjalan dengan baik dan benar. Pengujian mengelola data jalan akan dijabarkan sebagai berikut:

- Fungsi tambah data jalan

Fungsi tambah data jalan merupakan fungsi untuk menabahkan data tentang jalan ke *database* sistem yang dilakukan oleh *admin* jalan. Pengujian fungsi tambah data jalan akan dijelaskan pada Tabel 2.

Tabel 2 Pengujian fungsi tambah data jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah data jalan	Sesuai	Valid
Mengisi sebagian <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi	Sesuai	Valid
Mengisi <i>field</i> semua pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data jalan dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data jalan

Fungsi edit data jalan merupakan fungsi untuk merubah data tentang jalan untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* jalan. Pengujian fungsi edit data jalan akan dijelaskan pada Tabel 3.

Tabel 3 Pengujian fungsi edit data jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit data jalan, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i>	Proses edit data berhasil kemudian mengarahkan ke halaman menu data jalan dan	Sesuai	Valid

edit data, kemudian memilih tombol simpan.	menampilkan pesan data berhasil dirubah.		
--	--	--	--

- Fungsi hapus data jalan

Fungsi hapus data jalan merupakan fungsi untuk menghapus data jalan pada *database* sistem yang dilakukan oleh *admin* jalan. Pengujian fungsi hapus data jalan akan dijelaskan pada Tabel 4.

Tabel 4 Pengujian fungsi hapus data jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data jalan.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data jalan.	Sesuai	Valid

- e. Pengujian fungsi fitur mengelola data kondisi jalan

Pengujian mengelola data kondisi jalan dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data kondisi jalan sudah berjalan dengan baik dan benar. Pengujian mengelola data kondisi jalan akan dijabarkan sebagai berikut:

- Fungsi tambah data kondisi jalan

Fungsi tambah data kondisi jalan merupakan fungsi untuk menambahkan data tentang kondisi jalan ke *database* sistem yang dilakukan oleh *admin* jalan. Pengujian fungsi tambah data kondisi jalan akan dijelaskan pada Tabel 5.

Tabel 5 Pengujian fungsi tambah data kondisi jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul pop up untuk jumlah data yang akan ditambah pada tahun yang sama.	Sesuai	Valid
Pada pop up memasukan jumlah angka dan menekan tombol apply.	Mengarahkan ke <i>form</i> tambah data dalam bentuk tabel dengan baris sebanyak jumlah angka yang dimasukan.	Sesuai	Valid
Mengisi sebagian <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data kondisi jalan dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data kondisi jalan

Fungsi edit data jalan merupakan fungsi untuk merubah data tentang kondisi jalan untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* jalan. Pengujian fungsi edit data kondisi jalan akan dijelaskan pada Tabel 6.

Tabel 6 Pengujian fungsi edit data kondisi jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul <i>pop up form</i> edit data kondisi jalan, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid

Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid
--	--	--------	-------

- Fungsi hapus data kondisi jalan

Fungsi hapus data kondisi jalan merupakan fungsi untuk menghapus data kondisi jalan pada *database* sistem yang dilakukan oleh *admin* jalan. Pengujian fungsi hapus data kondisi jalan akan dijelaskan pada Tabel 7.

Tabel 7 Pengujian fungsi hapus data jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data kondisi jalan.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data kondisi jalan.	Sesuai	Valid

- f. Pengujian fitur laporan tahunan *admin* jalan

Pengujian fitur laporan tahunan *admin* jalan dilakukan untuk menguji apakah fungsi yang berjalan pada halaman menu Laporan untuk *admin* jalan sudah berjalan dengan baik dan benar. Pengujian fitur laporan tahunan *admin* jalan akan dijelaskan pada Tabel 8.

Tabel 8 Pengujian fitur laporan tahunan admin jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih rentang tahun untuk mengfilter data pengaduan kerusakan	Menampilkan dalam bentuk grafik garis dan tabel total pengaduan yang dilakukan	Sesuai	Valid

jalan, kemudian menekan tombol filter.	berdasarkan rentang tahun yang dipilih.		
Memilih tahun untuk mengfilter data kondisi jalan, kemudian tekan tombol filter.	Menampilkan dalam bentuk grafik <i>pie</i> dan tabel total presentasi kondisi jalan pada tahun yang dipilih.	Sesuai	Valid

g. Pengujian fungsi fitur mengelola data papan nama jalan

Pengujian mengelola data papan nama jalan dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data papan nama jalan sudah berjalan dengan baik dan benar. Pengujian mengelola data papan nama jalan akan dijabarkan sebagai berikut:

- Fungsi tambah data papan nama jalan

Fungsi tambah data papan nama jalan merupakan fungsi untuk menambahkan data papan nama jalan ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data papan nama jalan akan dijelaskan pada Tabel 9.

Tabel 9 Pengujian fungsi tambah data papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul <i>pop up form</i> tambah data papan nama jalan	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data papan nama jalan

Fungsi edit data papan nama jalan merupakan fungsi untuk merubah data papan nama jalan untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel.

Pengujian fungsi edit data papan nama jalan akan dijelaskan pada Tabel 10.

Tabel 10 Pengujian fungsi edit data papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul <i>pop up form</i> edit data papan nama jalan, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus data papan nama jalan

Fungsi hapus data papan nama jalan merupakan fungsi untuk menghapus data papan nama jalan pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi

hapus data papan nama jalan akan dijelaskan pada Tabel 11.

Tabel 11 Pengujian fungsi hapus data papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data papan nama jalan.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data papan nama jalan.	Sesuai	Valid

h. Pengujian fungsi fitur mengelola data pemasangan papan nama jalan

Pengujian mengelola data pemasangan papan nama jalan dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data pemasangan papan nama jalan sudah berjalan dengan baik dan benar. Pengujian mengelola data pemasangan papan nama jalan akan dijabarkan sebagai berikut:

- Fungsi tambah data pemasangan papan nama jalan

Fungsi tambah data pemasangan papan nama jalan merupakan fungsi untuk menambahkan data tentang pemasangan papan nama jalan ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data pemasangan papan nama jalan akan dijelaskan pada Tabel 12.

Tabel 12 Pengujian fungsi tambah data pemasangan papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah data pemasangan papan nama jalan.	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data pemasangan papan nama jalan dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data pemasangan papan nama jalan

Fungsi edit data pemasangan papan nama jalan merupakan fungsi untuk merubah data tentang pemasangan papan nama jalan untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel. Pengujian fungsi edit data pemasangan papan nama jalan akan dijelaskan pada Tabel 13.

Tabel 13 Pengujian fungsi edit data pemasangan papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit data pemasangan papan nama jalan, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian mengarahkan ke halaman menu data pemasangan papan nama jalan dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus data pemasangan papan nama jalan

Fungsi hapus data pemasangan papan nama jalan merupakan fungsi untuk menghapus data pemasangan papan nama jalan pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi hapus data pemasangan papan nama jalan akan dijelaskan pada Tabel 14.

Tabel 14 Pengujian fungsi hapus data pemasangan papan nama jalan.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data pemasangan papan nama jalan.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data pemasangan papan nama jalan.	Sesuai	Valid

i. Pengujian fungsi fitur mengelola data jenis rambu

Pengujian mengelola data jenis rambu dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data jenis rambu sudah berjalan dengan baik dan benar. Pengujian mengelola data jenis rambu akan dijabarkan sebagai berikut:

- Fungsi tambah data jenis rambu

Fungsi tambah data jenis rambu merupakan fungsi untuk menambahkan data jenis rambu ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data jenis rambu akan dijelaskan pada Tabel 15.

Tabel 15 Pengujian fungsi tambah data jenis rambu.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul <i>pop up form</i> tambah data jenis rambu	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data jenis rambu

Fungsi edit data jenis rambu merupakan fungsi untuk merubah data jenis rambu untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel. Pengujian fungsi edit data jenis rambu akan dijelaskan pada Tabel 16.

Tabel 16 Pengujian fungsi edit data jenis rambu.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul <i>pop up form</i> edit data jenis rambu, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus data jenis rambu

Fungsi hapus data jenis rambu merupakan fungsi untuk menghapus data jenis rambu pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi hapus data jenis rambu akan dijelaskan pada Tabel 17.

Tabel 17 Pengujian fungsi hapus data jenis rambu.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data jenis rambu.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data jenis rambu.	Sesuai	Valid

- j. Pengujian fungsi fitur mengelola data pemasangan rambu lalu lintas

Pengujian mengelola data pemasangan rambu lalu lintas dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data pemasangan rambu lalu lintas

sudah berjalan dengan baik dan benar. Pengujian mengelola data pemasangan rambu lalu lintas akan dijabarkan sebagai berikut:

- Fungsi tambah data pemasangan rambu lalu lintas

Fungsi tambah data pemasangan rambu lalu lintas merupakan fungsi untuk menabahkan data tentang pemasangan rambu lalu lintas ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data pemasangan rambu lalu lintas akan dijelaskan pada Tabel 18.

Tabel 18 Pengujian fungsi tambah data pemasangan rambu lalu lintas.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah data pemasangan rambu lalu lintas	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data pemasangan rambu lalu lintas dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data pemasangan rambu lalu lintas

Fungsi edit data pemasangan rambu lalu lintas merupakan fungsi untuk merubah data tentang pemasangan rambu lalu lintas untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel. Pengujian fungsi edit data pemasangan rambu lalu lintas akan dijelaskan pada Tabel 19.

Tabel 19 Pengujian fungsi edit data pemasangan rambu lalu lintas.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit data pemasangan rambu lalu lintas, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian mengarahkan ke halaman menu data pemasangan rambu lalu lintas dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus data pemasangan rambu lalu lintas

Fungsi hapus data pemasangan rambu lalu lintas merupakan fungsi untuk menghapus data pemasangan rambu lalu lintas pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi hapus data pemasangan rambu lalu lintas akan dijelaskan pada Tabel 20.

Tabel 20 Pengujian fungsi hapus data pemasangan rambu lalu lintas.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data pemasangan rambu lalu lintas.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data pemasangan rambu lalu lintas.	Sesuai	Valid

k. Pengujian fungsi fitur mengelola data pemasangan APILL

Pengujian mengelola data pemasangan APILL dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data pemasangan APILL sudah berjalan dengan baik dan benar. Pengujian mengelola data pemasangan APILL akan dijabarkan sebagai berikut:

- Fungsi tambah data pemasangan APILL

Fungsi tambah data pemasangan APILL merupakan fungsi untuk menambahkan data tentang pemasangan APILL ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data pemasangan APILL akan dijelaskan pada Tabel 21.

Tabel 21 Pengujian fungsi tambah data pemasangan APILL.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah data pemasangan APILL.	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data pemasangan APILL dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data pemasangan APILL

Fungsi edit data pemasangan APILL merupakan fungsi untuk merubah data tentang pemasangan APILL untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel. Pengujian fungsi edit data pemasangan APILL akan dijelaskan pada Tabel 22.

Tabel 22 Pengujian fungsi edit data pemasangan APILL.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit data pemasangan APILL, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid
Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian mengarahkan ke halaman menu data pemasangan APILL dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid

- Fungsi hapus data pemasangan APILL

Fungsi hapus data pemasangan APILL merupakan fungsi untuk menghapus data pemasangan APILL pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi hapus data pemasangan APILL akan dijelaskan pada Tabel 23.

Tabel 23 Pengujian fungsi hapus data pemasangan APILL.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data pemasangan APILL.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data pemasangan APILL.	Sesuai	Valid

1. Pengujian fungsi fitur mengelola data pemasangan RPPJ

Pengujian mengelola data pemasangan RPPJ dilakukan untuk menguji apakah fungsi untuk menambahkan, menghapus, dan merubah data pemasangan RPPJ sudah berjalan

dengan baik dan benar. Pengujian mengelola data pemasangan RPPJ akan dijabarkan sebagai berikut:

- Fungsi tambah data pemasangan RPPJ

Fungsi tambah data pemasangan RPPJ merupakan fungsi untuk menabahkan data tentang pemasangan RPPJ ke *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi tambah data pemasangan RPPJ akan dijelaskan pada Tabel 24.

Tabel 24 Pengujian fungsi tambah data pemasangan RPPJ.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol tambah data.	Muncul halaman <i>form</i> tambah data pemasangan RPPJ.	Sesuai	Valid
Mengisi sebagian <i>field</i> atau tidak memilih gambar pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Memunculkan pesan <i>error</i> “ <i>Please fill out this field</i> ” pada <i>field</i> yang tidak diisi atau pesan <i>error</i> “ <i>Please select a file</i> ” pada tombol <i>Upload a Image</i>	Sesuai	Valid
Mengisi semua <i>field</i> pada <i>form</i> tambah data, kemudian memilih tombol simpan.	Proses tambah data berhasil kemudian mengarahkan ke halaman menu data pemasangan RPPJ dan menampilkan pesan data berhasil disimpan.	Sesuai	Valid

- Fungsi edit data pemasangan RPPJ

Fungsi edit data pemasangan RPPJ merupakan fungsi untuk merubah data tentang pemasangan RPPJ untuk memperbaiki data sebelumnya yang dilakukan oleh *admin* faskel. Pengujian fungsi edit data pemasangan RPPJ akan dijelaskan pada Tabel 25.

Tabel 25 Pengujian fungsi edit data pemasangan RPPJ.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol edit data.	Muncul halaman <i>form</i> edit data pemasangan RPPJ, beserta menampilkan data yang akan dirubah ke <i>field</i> pada <i>form</i> .	Sesuai	Valid

Mengganti isi <i>field</i> yang akan dirubah pada <i>form</i> edit data, kemudian memilih tombol simpan.	Proses edit data berhasil kemudian mengarahkan ke halaman menu data pemasangan RPPJ dan menampilkan pesan data berhasil dirubah.	Sesuai	Valid
--	--	--------	-------

- Fungsi hapus data pemasangan RPPJ

Fungsi hapus data pemasangan RPPJ merupakan fungsi untuk menghapus data pemasangan RPPJ pada *database* sistem yang dilakukan oleh *admin* faskel. Pengujian fungsi hapus data pemasangan RPPJ akan dijelaskan pada Tabel 26.

Tabel 26 Pengujian fungsi hapus data pemasangan RPPJ.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih tombol hapus data.	Menampilkan <i>pop up</i> konfirmasi penghapusan data.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Ya.	Data akan dihapus dari <i>databases</i> dan tidak terlihat lagi dalam tabel data pemasangan RPPJ.	Sesuai	Valid
Pada <i>pop up</i> konfirmasi penghapusan data dipilih opsi Batal.	Data tidak akan dihapus dari <i>databases</i> dan masih terlihat dalam tabel data pemasangan RPPJ.	Sesuai	Valid

m. Pengujian fitur laporan tahunan *admin* faskel

Pengujian fitur laporan tahunan *admin* faskel dilakukan untuk menguji apakah fungsi yang berjalan pada halaman menu Laporan untuk *admin* faskel sudah berjalan dengan baik dan benar. Pengujian fitur laporan tahunan *admin* faskel akan dijelaskan pada Tabel 27.

Tabel 27 Pengujian fitur laporan tahunan admin faskel.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Memilih rentang tahun untuk mengfilter data pengaduan kerusakan	Menampilkan dalam bentuk grafik garis dan tabel total pengaduan dan juga	Sesuai	Valid

dan pemasangan fasilitas kelengkapan jalan, kemudian menekan tombol filter.	pemasangan fasilitas kelengkapan jalan yang dilakukan berdasarkan rentang tahun yang dipilih.		
---	---	--	--

2. Pengujian aplikasi untuk masyarakat

Pengujian aplikasi *mobile* untuk masyarakat menggunakan metode *black box* di lakukan pada beberapa fitur yaitu *login*, *sign up*, melihat kondisi jalan dan kelengkapan jalan, dan mengirim pengaduan kerusakan.

a. Pengujian fitur *login* aplikasi *mobile*

Pengujian fitur *login* aplikasi *mobile* dilakukan untuk menguji apakah fungsi yang berjalan pada halaman *login* sudah berjalan dengan baik dan benar. Pengujian fitur *login* akan dijelaskan pada Tabel 28.

Tabel 28 Pengujian black box fungsi *login* aplikasi *mobile*.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
<i>Form username</i> dan <i>password</i> di isi dengan data yang tidak sesuai.	Proses <i>login</i> tidak berhasil dan memunculkan pesan <i>error</i> .	Sesuai	Valid
<i>Form username</i> dan <i>password</i> di isi dengan data yang sesuai.	Proses <i>login</i> berhasil dan di arahkan ke halaman menu Home.	Sesuai	Valid

Pada pengujian Tabel 28 terdapat beberapa varian dalam input data yaitu apabila *form login* diisi dengan data tidak sesuai, maka *login* tidak berhasil dan akan memunculkan pesan *error*. Sedangkan bila *form login* diisi dengan data sesuai dengan data yang tersimpan pada *database*, maka admin berhasil *login* dan diarahkan ke halaman menu Home.

b. Pengujian fitur *signup* aplikasi *mobile*

Pengujian fitur *signup* aplikasi *mobile* dilakukan untuk menguji apakah fungsi yang berjalan pada halaman *signup* sudah berjalan dengan baik dan benar. Pengujian fitur *signup* akan dijelaskan pada Tabel 29.

Tabel 29 Pengujian black box fungsi *signup* aplikasi *mobile*.

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Mengisi form lengkap dan <i>username</i> yang digunakan sudah terdaftar.	Proses <i>signup</i> tidak berhasil dan memunculkan pesan <i>error username</i> telah digunakan.	Sesuai	Valid
Mengisi form lengkap dan <i>username</i> yang digunakan belum terdaftar.	Proses <i>signup</i> berhasil dan menampilkan pesan berhasil daftar akun.	Sesuai	Valid

Pada pengujian Tabel 29 terdapat beberapa varian dalam *input* data yaitu apabila *username* yang dimasukan sudah terdaftar sebelumnya, maka proses membuat akun tidak berhasil dan akan memunculkan pesan *error*. Sedangkan bila *username* yang dimasukan belum terdaftar sebelumnya pada *database*, maka masyarakat berhasil membuat akun untuk *login* ke aplikasi *mobile*.