

**APLIKASI SISTEM JEMPUT SAMPAH BERBASIS  
ANDROID UNTUK RUMAH KOS DAN  
AREA SEKITAR KAMPUS**



Disusun Oleh:

Nama : Ar Razy Fathan Rabbani

NIM : 17523163

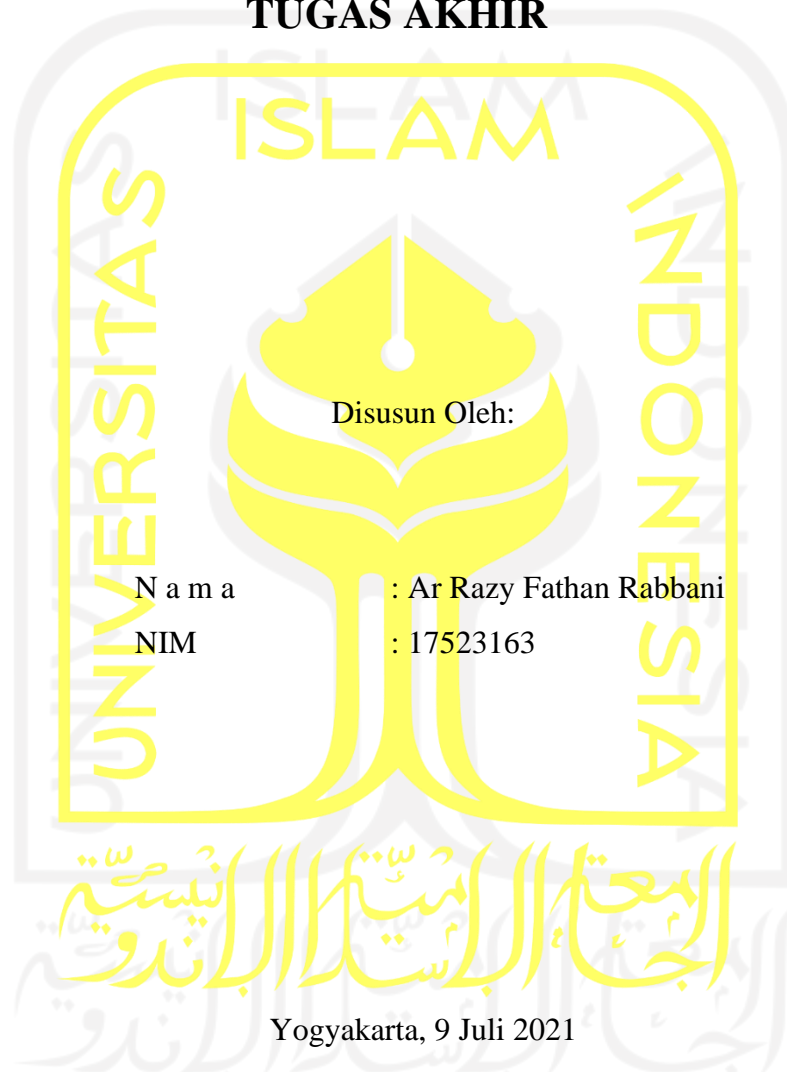
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA**

**2021**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**APLIKASI SISTEM JEMPUT SAMPAH BERBASIS  
ANDROID UNTUK RUMAH KOS DAN  
AREA SEKITAR KAMPUS**

**TUGAS AKHIR**



Pembimbing,

(Ahmad Raf'ie Pratama, S.T., MIT., Ph.D.)

**HALAMAN PENGESAHAN DOSEN PENGUJI**

**APLIKASI SISTEM JEMPUT SAMPAH BERBASIS  
ANDROID UNTUK RUMAH KOS DAN  
AREA SEKITAR KAMPUS  
TUGAS AKHIR**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Teknik Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 12 Agustus 2021

Tim Penguji

Ahmad Raf'ie Pratama, S.T., MIT., Ph.D.

**Anggota 1**

Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.

**Anggota 2**

Andhika Giri Persada, S.Kom., M.Eng.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

**HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR**

Yang bertanda tangan di bawah ini:

Nama : Ar Razy Fathan Rabbani

NIM : 17523163

Tugas akhir dengan judul:

**APLIKASI SISTEM JEMPUT SAMPAH BERBASIS  
ANDROID UNTUK RUMAH KOS DAN  
AREA SEKITAR KAMPUS**

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

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

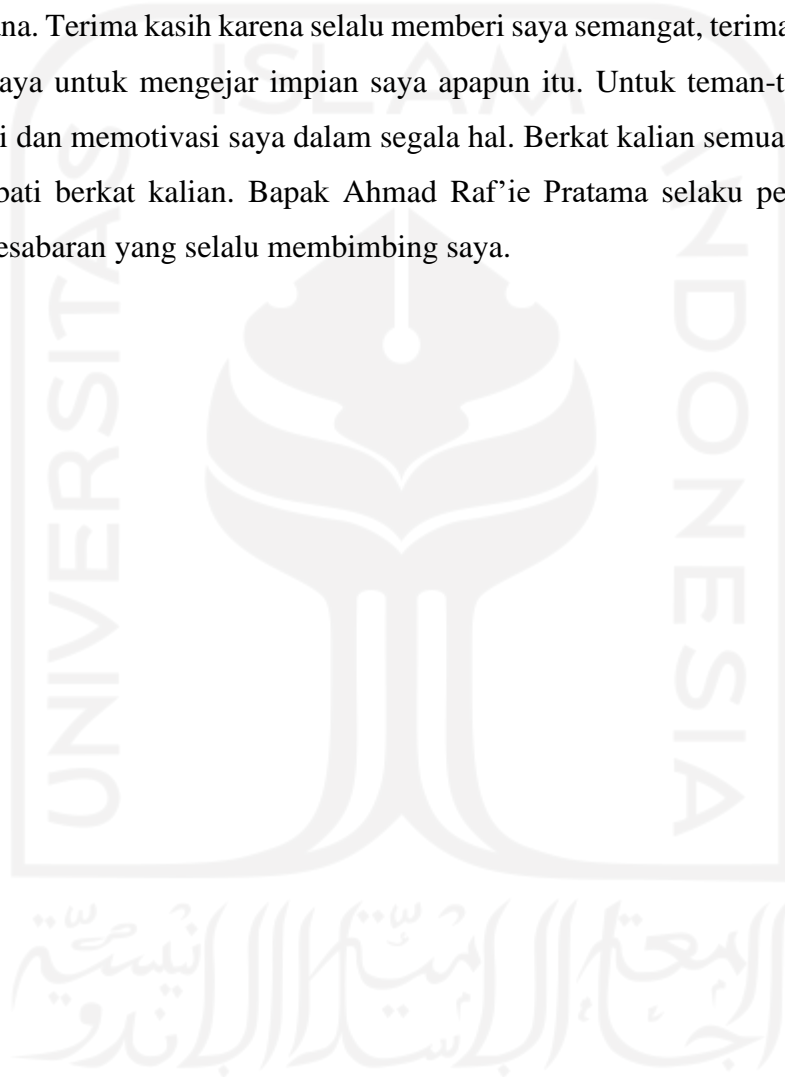
Yogyakarta, 10 Juli 2021



METERAI  
TEMPEL  
12-F75AJX277746434  
(Ar Razy Fathan Rabbani)

## HALAMAN PERSEMBAHAN

*Alhamdulillahillobbil'amin*, puji dan syukur saya panjatkan kepada Allah SWT dan Nabi Muhammad SAW atas segala limpahan karunianya yang telah diberikan, saya dapat menyelesaikan Tugas Akhir (TA) saya dengan lancar tanpa hambatan. Tugas akhir ini saya persembahkan untuk orang-orang yang sudah mendukung saya dan selalu memberikan semangat. Kedua orang tua saya, karena berkat beliau saya dapat berkesempatan menempuh pendidikan sarjana. Terima kasih karena selalu memberi saya semangat, terima kasih juga telah membebaskan saya untuk mengejar impian saya apapun itu. Untuk teman-teman saya yang selalu menemani dan memotivasi saya dalam segala hal. Berkat kalian semua rasa lelah, letih, bosan bisa terobati berkat kalian. Bapak Ahmad Raf'ie Pratama selaku pembimbing saya, dengan penuh kesabaran yang selalu membimbing saya.



## HALAMAN MOTO

“Sedikit lebih beda, lebih baik, dari pada sedikit lebih baik”

(Pandji Pragiwaksono)



## KATA PENGANTAR

Alhamdulillah, penulis panjatkan puji dan syukur atas kehadiran Allah SWT yang telah memberikan rahmat hidayah, serta karunia-Nya, sehingga laporan Tugas Akhir yang berjudul **Aplikasi Sistem Jemput Sampah Berbasis Android untuk Rumah Kos dan Area Sekitar Kampus** dapat selesai dengan lancar. Laporan tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata-1 (S1) Jurusan Informatika, Fakultas Teknologi Industri di Universitas Islam Indonesia dan sebagai bukti bahwa ilmu-ilmu yang telah dipelajari selama perkuliahan dapat diterapkan secara baik.

Penulis menyadari bahwa dalam penulisan laporan tugas akhir ini tidak lepas dari banyak sekali doa, dukungan, bimbingan yang senantiasa mengiringi dalam proses penyusunan laporan tugas akhir ini. Maka dari itu penulis mengucapkan terima kasih sebesar-besarnya kepada:

1. Kedua orang tua, atas segala pengorbanan dan doa yang telah diberikan kepada penulis.
2. Bapak Fathul Wahid, ST., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia
3. Bapak Hari Purnomo, Prof., Dr., Ir., M.T., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
4. Bapak Hendrik, S.T., M.Eng., selaku Ketua Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Program Studi Informatika – Program Sarjana.
6. Bapak Ahmad Munasir Raf'ie Pratama, S.T., MIT., Ph.D., selaku dosen pembimbing yang telah sabar dalam membimbing, mengarahkan dan memberikan masukan serta saran kepada penulis dalam menyelesaikan tugas akhir ini.
7. Segenap keluarga besar teman-teman di Fakultas Teknologi Industri terutama dari Jurusan Informatika Universitas Islam Indonesia yang telah memberikan bantuan dan dukungannya.
8. Semua pihak yang telah banyak membantu penulis dalam pelaksanaan Tugas Akhir yang tidak bisa disebutkan satu persatu.

Sebagai seorang yang terus ingin berkembang, penulis menyadari bahwa masih ada banyak kekurangan dalam penulisan dan penyusunan tugas akhir ini. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun untuk perbaikan di masa mendatang.

Semoga apa yang telah dihasilkan dapat memberikan manfaat untuk pembaca ataupun masyarakat.

Yogyakarta, 10 Juli 2021



(Ar Razy Fathah Rabbani)





## SARI

Permasalahan sampah di Indonesia merupakan sesuatu yang amat serius. Salah satu faktor utamanya adalah kurangnya kesadaran masyarakat dalam membuang dan mengelola sampah dengan benar. Tak terkecuali di area sekitar kampus dan rumah kos yang kebanyakan penghuninya adalah mahasiswa yang tak jarang kurang begitu peduli akan kebersihan lingkungan. Penelitian ini dilakukan untuk memberikan solusi alternatif bagi masyarakat, khususnya yang tinggal di rumah kos dan area sekitar kampus dalam melakukan pengelolaan dan pembuangan sampah yang baik dan benar dengan lebih mudah melalui sebuah aplikasi jemput sampah on-demand berbasis Android. Aplikasi ini dikembangkan dengan bahasa pemrograman Java di Android Studio dengan metode pengembangan *waterfall*. Untuk mengetahui titik penjemputan lokasi penjemputan sampah, aplikasi ini memanfaatkan layanan *MapBox Maps SDK* dan *MapBox Navigation SDK*. Berdasarkan hasil pengujian fungsionalitas, aplikasi telah berjalan dengan baik dan melewati seluruh skenario pengujian dengan metode *black box* dan dapat berjalan di ponsel pintar dengan sistem operasi Android 5.0 hingga 11, tanpa menutup kemungkinan di versi Android yang lebih baru.

Kata kunci: aplikasi Android, pembuangan sampah, jemput sampah, *on-demand*, lokasi penjemputan.

## GLOSARIUM

- Platform* Sebuah perpaduan antara perangkat lunak dan perangkat keras untuk menjalankan suatu program.
- Mapbox Maps SDK* Seperangkat alat dan program perangkat lunak yang digunakan untuk membangun sebuah aplikasi dengan layanan berbentuk peta.
- Mapbox Navigation SDK* Seperangkat alat dan program perangkat lunak yang digunakan untuk membangun sebuah aplikasi dengan layanan berbentuk navigasi.



## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING .....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI .....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR .....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR .....	vii
SARI.....	ix
GLOSARIUM .....	x
DAFTAR ISI .....	xi
DAFTAR TABEL .....	xiv
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan Penelitian .....	2
1.4 Batasan Masalah .....	3
1.5 Manfaat Penelitian Penelitian diharapkan dapat memberikan manfaat kepada:....	3
1.6 Metode Penelitian .....	3
1.6.1 Studi literatur .....	3
1.6.2 Analisis Kebutuhan .....	3
1.6.3 Perancangan Sistem.....	4
1.6.4 Implementasi .....	4
1.6.5 Pengujian .....	4
1.7 Sistematika Penulisan .....	4
BAB II KAJIAN PUSTAKA .....	6
2.1 Tinjauan Pustaka .....	6
2.2 Landasan Teori.....	7
2.2.1 Android.....	7
2.2.2 <i>Fused Location Provider API</i> .....	8
2.2.3 Geocoder .....	8
2.2.4 MapBox Maps SDK .....	8

2.2.5	MapBox Navigation SDK .....	9
2.2.6	Firebase Realtime Database .....	10
2.2.7	Firebase Cloud Messaging .....	10
2.2.8	Cloud Storage for Firebase .....	11
2.2.9	Android Studio .....	11
2.2.10	BlackBox Testing .....	11
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>13</b>
3.1	Analisis Kebutuhan .....	13
3.1.1	Analisis Kebutuhan Masukan .....	14
3.1.2	Analisis Kebutuhan Proses .....	14
3.1.3	Analisis Kebutuhan Keluaran .....	15
3.2	Perancangan Sistem .....	15
3.2.1	Use Case Diagram .....	15
3.2.2	Activity Diagram .....	17
3.2.3	Perancangan Database .....	32
3.2.4	Perancangan Antarmuka .....	32
3.3	Implementasi .....	46
3.4	Pengujian .....	52
3.4.1	Pengujian Black Box .....	52
3.4.2	Pengujian Usabilitas .....	53
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>56</b>
4.1	Hasil Implementasi Sistem .....	56
4.1.1	Halaman Login .....	56
4.1.2	Halaman Daftar Akun .....	57
4.1.3	Halaman Utama Aplikasi untuk Masyarakat .....	57
4.1.4	Halaman Kategori Sampah .....	58
4.1.5	Halaman Daftar Sampah .....	59
4.1.6	Halaman Detail Sampah .....	60
4.1.7	Halaman Menunggu <i>Pick-up</i> .....	61
4.1.8	Halaman Transaksi Selesai .....	62
4.1.9	Halaman Transaksi .....	63
4.1.10	Halaman Detail Transaksi .....	64
4.1.11	Halaman Tips .....	65
4.1.12	Halaman Profil .....	66

4.1.13	Halaman Utama Aplikasi untuk Admin .....	67
4.1.14	Halaman Kelola Data Sampah .....	68
4.1.15	Halaman Kelola Data Petugas .....	69
4.1.16	Halaman Transaksi Belum Diproses .....	70
4.1.17	Halaman Permintaan Jemput Sampah Ditolak .....	70
4.1.18	Halaman Detail <i>Pick-up</i> .....	71
4.1.19	Halaman Utama Aplikasi untuk Petugas .....	72
4.1.20	Halaman Detail Permintaan Penjemputan .....	73
4.1.21	Halaman Rute Lokasi Penjemputan .....	74
4.1.22	Halaman Navigasi Menuju Lokasi Titik Penjemputan .....	75
4.2	Hasil Pengujian .....	76
4.2.1	Hasil Pengujian Black Box Aplikasi untuk masyarakat .....	76
4.2.2	Hasil Pengujian Black Box Aplikasi untuk Admin .....	77
4.2.3	Hasil Pengujian Black Box Aplikasi untuk Petugas .....	78
4.2.4	Hasil Pengujian Usabilitas .....	79
BAB V KESIMPULAN DAN SARAN .....		82
5.1	Kesimpulan .....	82
5.2	Saran .....	82
DAFTAR PUSTAKA .....		83
LAMPIRAN .....		85

**DAFTAR TABEL**

Tabel 3.1 Indikator penilaian pengujian usabilitas .....	53
Tabel 3.2 Rancangan daftar pertanyaan kuesioner .....	54
Tabel 3.3 Keterangan nilai pengujian usabilitas .....	55
Tabel 4.1 Hasil pengujian <i>black box</i> aplikasi untuk masyarakat.....	77
Tabel 4.2 Hasil pengujian <i>black box</i> aplikasi untuk admin. ....	78
Tabel 4.3 Hasil pengujian <i>black box</i> aplikasi untuk petugas. ....	78
Tabel 4.4 Hasil pengujian usabilitas .....	80



## DAFTAR GAMBAR

Gambar 3.1 Use case diagram .....	16
Gambar 3.2 Activity Diagram daftar akun baru. ....	17
Gambar 3.3 Activity Diagram login. ....	18
Gambar 3.4 Activity Diagram melihat daftar sampah. ....	19
Gambar 3.5 Activity Diagram melihat artikel. ....	20
Gambar 3.6 Activity Diagram melihat informasi akun. ....	21
Gambar 3.7 Activity Diagram input alamat. ....	22
Gambar 3.8 Activity Diagram melihat status penjemputan. ....	23
Gambar 3.9 Activity Diagram melihat detail penjemputan sampah. ....	24
Gambar 3.10 Activity Diagram menambahkan sampah keantrian. ....	25
Gambar 3.11 Activity Diagram mengirim permintaan jemput sampah. ....	26
Gambar 3.12 Activity Diagram kelola data sampah. ....	27
Gambar 3.13 Activity Diagram memproses permintaan jemput sampah. ....	28
Gambar 3.14 Activity Diagram kelola data petugas. ....	29
Gambar 3.15 Activity Diagram memproses melihat rute penjemputan sampah. ....	30
Gambar 3.16 Activity Diagram navigasi ke titik penjemputan sampah. ....	31
Gambar 3.17 Skema data JSON. ....	32
Gambar 3.18 Rancangan halaman login. ....	33
Gambar 3.19 Rancangan halaman daftar akun. ....	34
Gambar 3.20 Rancangan halaman utama untuk masyarakat. ....	34
Gambar 3.21 Rancangan halaman kategori sampah. ....	35
Gambar 3.22 Rancangan halaman daftar sampah. ....	36
Gambar 3.23 Rancangan halaman detail sampah. ....	36
Gambar 3.24 Rancangan halaman menunggu <i>pick-up</i> . ....	37
Gambar 3.25 Rancangan halaman transaksi selesai. ....	38
Gambar 3.26 Rancangan halaman transaksi. ....	38
Gambar 3.27 Rancangan halaman transaksi. ....	39
Gambar 3.28 Rancangan halaman tips. ....	40
Gambar 3.29 Rancangan halaman profil. ....	40
Gambar 3.30 Rancangan halaman utama aplikasi untuk admin. ....	41
Gambar 3.31 Rancangan halaman kelola data sampah. ....	41
Gambar 3.32 Rancangan halaman kelola data petugas. ....	42

Gambar 3.33 Rancangan halaman transaksi belum diproses.....	43
Gambar 3.34 Rancangan halaman permintaan jemput sampah ditolak.....	43
Gambar 3.35 Rancangan halaman detail <i>pick-up</i> . ....	44
Gambar 3.36 Rancangan halaman utama aplikasi untuk petugas.....	44
Gambar 3.37 Rancangan halaman detail permintaan penjemputan.....	45
Gambar 3.38 Rancangan halaman rute lokasi penjemputan.....	45
Gambar 3.39 Rancangan halaman navigasi menuju lokasi titik penjemputan. ....	46
Gambar 3.40 Kode program menambahkan permissions.....	47
Gambar 3.41 Kode program mengetahui lokasi pengguna.....	47
Gambar 3.42 Kode program penggunaan geocoder. ....	48
Gambar 3.43 Menambahkan akses token. ....	48
Gambar 3.44 Menambahkan library MapBox SDK.....	49
Gambar 3.45 Kode program menampilkan peta <i>MapBox Maps</i> .....	49
Gambar 3.46 Kode program mengatur komponen pada peta. ....	50
Gambar 3.47 Kode program <i>method</i> <i>getRoute()</i> .....	51
Gambar 3.48 Kode program menampilkan navigasi MapBox SDK. ....	52
Gambar 4.1 Halaman login.....	56
Gambar 4.2 Halaman daftar akun. ....	57
Gambar 4.3 Halaman menu utama pada aplikasi masyarakat. ....	58
Gambar 4.4 Halaman kategori sampah.....	59
Gambar 4.5 Halaman daftar sampah.....	60
Gambar 4.6 Halaman detail sampah. ....	61
Gambar 4.7 Halaman menunggu <i>pick-up</i> . ....	62
Gambar 4.8 Halaman transaksi selesai. ....	63
Gambar 4.9 Halaman transaksi.....	64
Gambar 4.10 Halaman detail transaksi. ....	65
Gambar 4.11 Halaman tips. ....	66
Gambar 4.12 Halaman profil. ....	67
Gambar 4.13 Halaman utama aplikasi untuk admin.....	68
Gambar 4.14 Halaman kelola data sampah.....	69
Gambar 4.15 Halaman kelola data petugas.....	69
Gambar 4.16 Halaman kelola data petugas.....	70
Gambar 4.17 Halaman permintaan jemput sampah ditolak.....	71
Gambar 4.18 Halaman detail <i>pick-up</i> . ....	72



Gambar 4.19 Halaman halaman utama aplikasi untuk petugas. ....	73
Gambar 4.20 Halaman detail permintaan penjemputan.....	74
Gambar 4.21 Halaman rute lokasi penjemputan.....	75
Gambar 4.22 Halaman navigasi menuju lokasi titik penjemputan. ....	76



## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pengelolaan sampah di Indonesia masih cukup jauh dari kata baik, hal ini masih terjadi karena kurangnya kesadaran masyarakat terhadap pengelolaan dan pengolahan sampah yang ada di sekitar lingkungan masyarakat. Tidak hanya itu, peran pemerintah dalam pengelolaan sampah di Indonesia juga masih kurang. Indonesia diperkirakan menghasilkan 64 juta ton sampah setiap tahunnya. Merujuk data dari *Sustainable Waste Indonesia* (SWI) pada tahun 2017, dari angka tersebut baru 7% yang didaur ulang, sementara 69% diantaranya menumpuk di tempat pembuangan akhir (TPA). Lebih parahnya lagi 24% sisanya dibuang sembarangan dan mencemari lingkungan (Tim Publikasi Katadata, 2019). Kemudian berdasarkan data dari Kementerian Lingkungan Hidup dan Kehutanan (KLHK), total sampah di Indonesia tahun 2019 akan mencapai 68 juta ton dan sampah plastic diperkirakan akan mencapai 9,52 juta ton (Caca, 2020).

Yang menjadikan fokus utama adalah kesadaran masyarakat dalam membuang sampah. Masih banyak masyarakat yang malas membuang sampah dan memilih membuang sampah tidak pada tempatnya. Masyarakat sering kali membuang sampah secara sembarangan, seperti membuang sampah di sungai, di selokan, di pinggir jalan dan lainnya. Lingkungan rumah kos dan area kampus menjadi salah satu lingkungan yang perlu diperhatikan dalam hal pengelolaan sampah. Keberadaan rumah kos memang memberikan manfaat bagi sebagian orang seperti mahasiswa yang sedang menempuh pendidikan maupun masyarakat dari luar kota yang ingin bekerja. Rumah kos juga memberikan dampak positif secara ekonomi karena dapat memberikan tambahan penghasilan bagi masyarakat sekitar. Namun rumah kos juga membawa dampak negatif yaitu keberadaan sampah di daerah sekitar yang terdapat rumah kos akan bertambah. Tidak hanya sekitar rumah kos saja yang akan terdampak penambahan sampah, area kampus yang berdekatan dengan rumah kos juga akan ikut terkena dampak dari penambahan keberadaan sampah.

Sampah adalah suatu bahan yang terbuang atau dibuang dari sumber hasil aktivitas manusia maupun proses alam yang belum memiliki nilai ekonomis (Istilah Lingkungan untuk Manajemen, Ecolink, 1996). Sampah digolongkan menjadi 2 kategori yaitu sampah organik dan anorganik. Sampah organik merupakan sampah yang mudah didaur ulang seperti daun-

daun kering, sisa sayuran, sisa makanan dan lainnya. Sedangkan sampah anorganik merupakan sampah yang tidak mudah terurai seperti plastik, kertas, botol, gelas, kaleng, sampah elektronik dan lainnya.

Sampah anorganik menjadi salah satu sampah yang susah dikelola atau diolah, karena susah terurai. Jumlahnya setiap harinya bertambah dan cenderung diabaikan keberadaannya oleh masyarakat. Sampah anorganik ini juga memiliki dampak buruk terhadap lingkungan dan makhluk hidup. Contohnya adalah adanya banjir yang disebabkan naiknya volume air sungai yang tersumbat oleh sampah, sehingga air tidak bisa mengalir dengan baik. Dan masih banyak dampak buruk yang ditimbulkan oleh sampah seperti mencemari tanah, mencemari air, menyebabkan polusi udara, menyebabkan berbagai penyakit, dan masih banyak lagi.

Dari permasalahan tersebut maka dari itu pada penelitian ini penulis mengusulkan solusi untuk mengembangkan sistem jemput sampah untuk rumah kos dan area kampus berupa aplikasi *mobile* berbasis Android sebagai solusi alternatif untuk membantu pengelolaan sampah. Dengan aplikasi ini masyarakat khususnya mahasiswa dan warga sekitar rumah kos dan area kampus bisa mengajukan permintaan penjemputan sampah sesuai lokasi titik penjemputan. Kemudian dari lokasi tersebut petugas bisa menjemput sampah sesuai dengan titik lokasi penjemputan. Diharapkan dengan aplikasi ini masyarakat lebih teredukasi dan lebih sadar terhadap pentingnya membuang sampah pada tempatnya agar tidak mencemari lingkungan dan mengganggu ekosistem lingkungan.

## **1.2 Rumusan Masalah**

Dari latar belakang di atas maka disusun rumusan masalah yang akan dibahas pada penelitian ini, berupa: Bagaimana aplikasi sistem jemput sampah berbasis android untuk rumah kos dan area sekitar kampus dikembangkan untuk memberikan alternatif bagi masyarakat dalam membuang sampah?

## **1.3 Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah sebagai berikut:

- a. Mengembangkan aplikasi sistem jemput sampah berbasis android untuk rumah kos dan sekitar area kampus untuk memberikan alternatif bagi masyarakat untuk membuang atau mengelola sampah dengan benar.

#### 1.4 Batasan Masalah

Dalam penyusunan penelitian ini, penulis akan membatasi ruang lingkup pembahasan supaya lebih terfokus pada permasalahan yang diangkat. Adapun ruang lingkup dalam penelitian ini adalah:

- a. Sistem yang dibangun hanya berjalan pada *smartphone* berbasis android dengan versi minimal 5.0.
- b. Sampah yang dapat dikirim oleh pengguna berupa sampah anorganik yang dapat didaur ulang.

#### 1.5 Manfaat Penelitian

Penelitian diharapkan dapat memberikan manfaat kepada:

- a. Penulis  
Penulis dapat menerapkan dan mengimplementasikan ilmu pengetahuan mengenai pengembangan sebuah aplikasi android dan teknologi-teknologi terkait yang digunakan dalam penelitian ini.
- b. Masyarakat  
Memberikan alternatif lain bagi masyarakat dalam pengelolaan sampah agar tidak membuang sampah secara sembarangan.

#### 1.6 Metode Penelitian

Metode penelitian adalah langkah-langkah yang dilakukan dalam penyusunan tugas akhir yang memiliki beberapa tahapan meliputi:

##### 1.6.1 Studi literatur

Pada tahapan studi literatur penulis mempelajari dan memahami mengenai beberapa penelitian yang berkaitan atau yang relevan untuk mendukung penyusunan penelitian ini. Sumber yang bisa dipakai bisa berasal dari buku, jurnal, *website*, internet ataupun karya ilmiah yang relevan dengan pengelolaan sampah dan pengembangan aplikasi berbasis android.

##### 1.6.2 Analisis Kebutuhan

Analisis kebutuhan dibutuhkan untuk mencari dan merumuskan apa saja kebutuhan sistem yang akan dikembangkan. Dengan melakukan analisis kebutuhan

sistem diharapkan kedepannya sistem yang akan dikembangkan memiliki kesesuaian dengan keinginan awal yang ingin dicapai.

### **1.6.3 Perancangan Sistem**

Tahapan perancangan dilakukan untuk memberikan gambaran dari sistem yang akan dikembangkan seperti siapa saja pengguna yang terkait, bagaimana interaksi pengguna dan sistem, perancangan database dan juga perancangan antarmuka aplikasi yang akan dibuat.

### **1.6.4 Implementasi**

Tahapan implementasi dilakukan untuk mengimplementasikan semua hasil perancangan yang telah dilakukan untuk dikembangkan menjadi sebuah aplikasi.

### **1.6.5 Pengujian**

Tahapan terakhir adalah pengujian, pada tahapan ini dilakukan pengujian untuk menguji perangkat lunak atau aplikasi yang telah dikembangkan dari segi fungsionalitas. Tujuannya adalah untuk memastikan bahwa perangkat lunak atau aplikasi yang telah dikembangkan dapat beroperasi dengan baik dan memenuhi kebutuhan pengguna.

## **1.7 Sistematika Penulisan**

Sistematika penulisan yang digunakan dalam penyusunan laporan ini adalah sebagai berikut:

### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab ini berisikan mengenai teori-teori yang menjadi acuan dan berkaitan dengan pengembangan aplikasi dan *review* beberapa penelitian terkait mengenai pengelolaan sampah dan pengembangan aplikasi.

### **BAB III METODOLOGI**

Bab ini berisikan mengenai tahapan-tahapan dalam pengembangan aplikasi muai dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi dan juga pengujian.

### **BAB IV HASIL DAN PEMBAHASAN**

Bab ini berisi mengenai pokok bahasan implementasi dari perancangan yang sudah dilakukan, hasil pengembangan aplikasi dan hasil pengujian aplikasi.

### **BAB V KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dan saran dari seluruh proses pengembangan aplikasi yang telah dilakukan.



## BAB II KAJIAN PUSTAKA

### 2.1 Tinjauan Pustaka

Beberapa penelitian sebelumnya mengenai pengelolaan sampah yang akan digunakan untuk melihat fitur apa yang dibutuhkan dalam pengembangan aplikasi pada penelitian ini dan untuk menentukan posisi penelitian ini, termasuk *platform* apa yang akan digunakan pada aplikasi yang akan dikembangkan. Penelitian yang pertama yang dilakukan oleh Jamaludin (2016), dalam penelitiannya topik bahasan utamanya adalah menganalisa dan merancang sistem berbasis Android yang dapat membantu pengelolaan sampah di kecamatan Rappocini. Melalui sistem tersebut keluhan masyarakat dapat tersalurkan dengan mudah melalui *smartphone*. Aplikasi tersebut memberikan informasi berupa gambar tumpukan sampah, lokasi detail, tanggal dan waktu pelaporan serta keluhan yang nantinya akan diinformasikan oleh masyarakat.

Penelitian kedua dilakukan oleh Firmansya et al. (2019), penelitiannya menghasilkan sebuah aplikasi Android untuk mempermudah pencarian informasi mengenai bank sampah di Kota Jambi. Dalam aplikasinya juga terdapat menu berita atau artikel seputar pengelolaan sampah dan juga fitur layanan jemput sampah.

Penelitian ketiga dilakukan oleh Aziz et al. (2018), penelitiannya membahas mengenai sebuah rancangan tampilan *user interface* aplikasi pengelolaan administrasi dan bisnis bank sampah. Fitur-fitur yang dirancang adalah fitur kelola data nasabah, fitur kelola data sampah, fitur kelola data transaksi tabungan sampah dan fitur kelola jadwal penjemputan sampah.

Penelitian keempat dilakukan oleh Masruroh et al. (2015), pada penelitian yang dilakukan menghasilkan sebuah aplikasi bank sampah untuk membantu Yayasan Bunga Melati Indonesia dan bank sampah Melati Bersih dalam perhitungan dan pengelolaan data. Fitur yang ada pada aplikasi berfokus pada perhitungan dan pengelolaan data transaksi pada bank sampah.

Penelitian yang kelima dilakukan oleh Hadarkho et al. (2016), hasil dari penelitiannya berupa sistem informasi manajemen bank sampah pada UKM di Yogyakarta. Tepatnya di Bank Sampah Gemah Ripah Badegan, Bantul. Sistem informasi yang dikembangkan digunakan untuk mengelola manajemen bank sampah seperti untuk mengelola data master, pengelolaan umum, pengelolaan penggajian, pengelolaan transaksi dan pengelolaan laporan.

Penelitian terakhir dilakukan oleh Magdalena et al. (2019), dalam penelitiannya menghasilkan sistem informasi pengelolaan pembayaran retribusi sampah berbasis web. Fitur yang dikembangkan yang ada di dalam sistem informasi ini diantaranya, fitur entri data kategori sampah, entri data kendaraan bidang pengelolaan sampah, transaksi absensi petugas bidang pengelolaan sampah, entri surat penunjuk bidang sampah, entri data wilayah, entri data pelanggan retribusi, transaksi entri data retribusi, transaksi entri bukti retribusi dan laporan retribusi. Dari penelitian-penelitian yang sudah disebutkan diatas dapat diketahui bahwa peran teknologi informasi bisa digunakan sebagai solusi alternatif untuk membantu pengelolaan sampah di Indonesia, salah satunya adalah aplikasi jemput sampah *on-demand* yang dapat memberikan kemudahan lebih bagi penggunanya.

Di sisi lain, penelitian terdahulu juga menunjukkan ada kecenderungan mahasiswa yang tinggal di rumah kos sebagai pendatang untuk kurang peduli akan kebersihan lingkungan sekitarnya (Kumurur, 2012). Oleh karena itu, perlu dikembangkan sebuah aplikasi jemput sampah yang ditargetkan kepada kelompok masyarakat ini, yakni mahasiswa yang tinggal di rumah kos dan area sekitar kampus. Sehubungan dengan itu pula, mengingat karakteristik usia para calon penggunanya, diperlukan adanya aplikasi yang memiliki antarmuka kekinian agar dapat menarik perhatian dan kesetiaan penggunanya. Selain itu, dikarenakan tingkat adopsi *platform* Android yang sangat tinggi di Indonesia, termasuk di kalangan mahasiswa dan akademisi, maka aplikasi Android perlu menjadi pilihan utama untuk pengembangan purwarupa awal sebelum diadopsi di *platform* lain semisal iOS. Hal ini lah yang menjadi fokus utama penelitian ini sekaligus yang membedakannya dengan penelitian-penelitian terdahulu berdasarkan hasil kajian literatur.

## **2.2 Landasan Teori**

### **2.2.1 Android**

Android merupakan salah satu sistem operasi yang sangat populer yang digunakan pada *smartphone*, *smartwatch*, *smart tv* dan perangkat lainnya. Sistem operasi yang berbasis linux ini menempati posisi pertama dalam *Mobile Operation System Market Share* di Indonesia per Februari 2021 dengan persentase 92.03%, sedangkan saingan terberatnya yaitu iOS berada di peringkat kedua dengan persentase 7.75% diikuti dengan Samsung dan Windows di peringkat ketiga dan keempat. (Statcounter Global Stat, 2021)

Karena kepopuleran Android, banyak sekali aplikasi yang dikembangkan pada perangkat android khususnya pada *smartphone*. Kemudahan dan lengkapnya dokumentasi yang diberikan



oleh Google terhadap pengembang membuat proses pengembangan aplikasi Android menjadi lebih mudah dan cepat. Setidaknya ada kurang lebih 88.5 ribu aplikasi yang dirilis via Google Play selama bulan Februari 2021 (Statista Research Department, 2021).

### **2.2.2 Fused Location Provider API**

*Fused Location Provider API* merupakan layanan yang diberikan oleh Google yang digunakan untuk mendapatkan informasi lokasi perangkat pengguna Android. Aplikasi yang dikembangkan dengan *Fused Location Provider API* bisa memanfaatkan sinyal yang diberikan oleh beberapa sensor di perangkat untuk menentukan lokasi perangkat. Informasi yang didapatkan melalui *Fused Location Provider API* salah satunya adalah lokasi saat ini yang direpresentasikan dengan titik koordinat *latitude* dan *longitude*. *Fused Location Provider API* memungkinkan perangkat bisa mendapatkan informasi lokasi terkini dari dan update lokasi secara berkala dari perangkat pengguna (Google Developers, n.d).

### **2.2.3 Geocoder**

Geocoder merupakan salah satu *class* yang digunakan dalam pengembangan Android untuk menangani proses *geocoding* dan *reverse geocoding*. *Geocoding* sendiri merupakan proses untuk mentransformasikan alamat rumah atau nama lokasi suatu tempat ke dalam titik koordinat *latitude* dan *longitude*. Sedangkan *reverse geocoding* kebalikan dari *geocoding* yang mentransformasikan titik koordinat *latitude* dan *longitude* menjadi nama sebuah lokasi yang berbentuk sebuah teks. Hasil dari *geocoding* yang dilakukan mungkin akan menghasilkan informasi dengan detail yang berbeda, ada yang menghasilkan informasi dengan alamat lengkap ada juga yang menghasilkan hanya berisi nama kota dan kode pos (Google Developers, n.d).

### **2.2.4 MapBox Maps SDK**

MapBox *Maps* SDK merupakan salah satu penyedia layanan terbesar yang digunakan untuk menampilkan peta yang dirancang khusus untuk situs *web* dan aplikasi *mobile*. Dengan menggunakan *mapbox maps* pengembang akan diberikan banyak opsi pilihan desain *maps* yang akan digunakan pada aplikasi Android. *Mapbox maps* juga dapat memberikan visualisasi data lebih cepat dan kontrol yang lebih lengkap. *Mapbox maps* juga dapat berjalan pada perangkat *cross-platform* yang artinya bisa digunakan pada *website*, Android dan juga iOS. (MapBox *Maps*, n.d)

Teknologi yang digunakan mapbox *maps* berbasis bahasa Node.js, Mapnik (perangkat *open-source* yang digunakan untuk merender *maps*), GDAL dan Leaflet (JavaScript *library* untuk peta interaktif). Mapbox *maps* menawarkan banyak *tools* yang membantu pengembang untuk mengintegrasikan layanan peta ke aplikasi yang mereka kembangkan. Berikut beberapa kelebihan yang bisa didapatkan ketika menggunakan layanan mapbox maps (Bulatovych, n.d.).

a. Pilihan kostumisasi yang unik

Mapbox menawarkan pilihan kustomisasi yang lebih banyak, mapbox memungkinkan pengembang dapat membuat gaya peta sesuai dengan keinginan dan kebutuhan pengembang. Pengembang dapat mengatur *font* dan skema warna serta menambahkan fungsionalitas seperti petunjuk arah *turn-by-turn* dan juga informasi medan pada peta.

b. *Open-source SDKs*

Mapbox maps merupakan *open source*, mapbox membagikan *source code* di GitHub sehingga memungkinkan pengembang lain dapat berkontribusi untuk melakukan peningkatan kualitas dari mapbox itu sendiri.

c. *Offline Maps*

Mapbox memberikan banyak fleksibilitas terkait mode *offline* nya. Aplikasi yang terintegrasi dengan mapbox SDK dapat mengunduh peta untuk area geografis yang dipilih untuk bisa digunakan dikemudian hari jika memang tidak memiliki akses internet.

Namun, ada satu kekurangan dari mapbox terutama pada layanan peta nya yaitu cangkupan peta yang relatif lebih lemah dari layanan penyedia *maps* lainnya seperti Google *Maps*. Ada banyak tempat di mana Google memiliki jangkauan yang lebih baik, alasannya adalah karena Google Maps terus diperbarui.

### 2.2.5 MapBox Navigation SDK

Mapbox *Navigation SDK* merupakan salah satu penyedia layanan navigasi yang bisa digunakan pada perangkat Android dan iOS. Navigasi SDK memungkinkan pengembang untuk menambahkan fitur pemandu arah secara *turn-by-turn* dalam aplikasi dengan menyediakan komponen *UI*, *core navigation logic* dan API yang menyediakan peta. Mapbox *navigation* mampu secara otomatis mencari rute tercepat meskipun kondisi yang ada di sepanjang jalan berubah-ubah. Dengan memanfaatkan teknologi *Enhanced Location Engine* navigasi mampu secara terus menerus memfilter dan memproses sinyal *GPS* dan memperkirakan lokasi pengemudi saat ini, bahkan ketika sinyal *GPS* tidak tersedia. Hal ini

memungkinkan Navigasi SDK memberikan akurasi posisi yang tinggi dengan latensi yang rendah. Sehingga lokasi pengemudi di peta tetap sinkron dengan lokasi sebenarnya (Mapbox Navigation, n.d.).

*Navigation SDK* juga menyediakan fitur *cinematic camera*, dengan fitur ini kamera pada komponen *UI* akan otomatis *zoom in* untuk melacak setiap manuver dengan tepat. Sama halnya dengan *mapbox maps*, *navigation SDK* juga dapat dikustomisasi dengan mudah oleh pengembang agar sesuai dengan kebutuhan pengembang. *Navigation SDK* juga menyediakan *offline routing*, ini memungkinkan pengguna dapat menggunakan navigasi ketika tidak tersambung dengan jaringan internet.

### 2.2.6 Firebase Realtime Database

*Firebase realtime database* merupakan *cloud-hosted NoSQL* database yang mampu menyimpan dan menyinkronkan data antara pengguna dan aplikasi secara *realtime*. Data yang disimpan dalam bentuk JSON yang disinkronisasi setiap kali terhubung ke internet. Sinkronisasi secara *realtime* memudahkan pengguna untuk mengakses data mereka dari perangkat apapun web ataupun *mobile* dan membantu pengguna berkolaborasi satu sama lain. Dengan *Firebase Realtime Database* memungkinkan pengembang dapat membuat aplikasi tanpa memerlukan server. Saat pengguna sedang *offline* atau tidak tersambung ke internet, *realtime database* menggunakan *cache* lokal yang ada di perangkat untuk menyajikan dan menyimpan perubahan. Saat perangkat *online* kembali data local tersebut akan otomatis disinkronkan. *Realtime database* menyediakan fleksibilitas berupa *expression-based rules language*, yang disebut *Firebase Realtime Database Security Rules*. Dengan aturan tersebut pengembang dapat menentukan bagaimana data disimpan dan kapan data dapat dibaca ataupun diubah. (Firebase Realtime Database, n.d.).

### 2.2.7 Firebase Cloud Messaging

*Firebase Cloud Messaging* (FCM) merupakan *cross-platform messaging* yang mampu mengirimkan pesan dari server ke perangkat pengguna tanpa biaya. Dengan *Firebase Cloud Messaging*, pengembang dapat mengirimkan notifikasi kepada pengguna bahwa terdapat email baru atau data lain sesuai dengan kebutuhan. Terdapat 3 cara untuk mendistribusikan pesan ke perangkat *client*, yaitu kesatu perangkat, ke sekelompok perangkat atau ke perangkat yang berlangganan ke suatu topik. *Firebase Cloud Messaging* menawarkan berbagai opsi dan

kemampuan pengiriman pesan. Dengan FCM pengembang dapat mengirimkan dua jenis pesan yaitu *notification message* dan *data message* (Firebase Cloud Messaging, n.d.).

### 2.2.8 Cloud Storage for Firebase

*Cloud Storage for Firebase* merupakan salah satu layanan dari Firebase yang digunakan untuk menyimpan dan menampilkan konten berupa file-file yang dapat diunggah oleh pengguna, file bisa berupa foto, audio ataupun video. *Cloud storage* mampu melakukan operasi yang stabil bagaimanapun jaringannya. Ini memungkinkan proses *download* dan *upload* oleh pengguna dapat berjalan dengan stabil. Konten yang didownload atau diupload akan dimulai kembali di saat proses *download* atau *upload* berhenti, tidak diulang dari awal. Sehingga dapat menghemat waktu dan *bandwith* pengguna. Cloud storage juga terintegrasi dengan *Firebase Authentication*, sehingga pengembang dapat menerapkan model keamanan sesuai dengan kebutuhan(Cloud Storage for Firebase, n.d.).

### 2.2.9 Android Studio

Android Studio merupakan IDE (*Integrated Development Enviroment*) resmi yang dikeluarkan Google, yang digunakan untuk mengembangkan aplikasi pada setiap tipe perangkat Android seperti *smartphone*, *AndroidTV*, *smartwatch* dan lain-lain. Android studio diluncurkan oleh Google pertama kali pada tahun 2013 pada acara konferensi Google I/O. Dengan adanya android studio, Google secara resmi menghentikan dukungan penuh terhadap Eclipse pada tahun 2016. Android studio menawarkan banyak kelebihan-kelebihan yang menunjang produktifitas ketika mengembangkan aplikasi android. Seperti sistem *build* dengan Gradle yang fleksibel, dengan Gradle pengembangan aplikasi akan semakin mudah dan cepat karena bayaknya fitur yang ada di Gradle itu sendiri. Android studio juga didukung oleh *emulator* yang cepat dan juga kaya akan fitur, yang tentunya sudah tersedia di dalam android studio. Dokumentasi penggunaan android studio juga sangat lengkap dan terstruktur, sehingga memudahkan pengembang jika terjadi kesulitan saat mengembangkan aplikasi dengan android studio.

### 2.2.10 BlackBox Testing

*Black Box testing* merupakan salah satu metode yang digunakan untuk pengujian suatu fungsionalitas sebuah aplikasi atau perangkat lunak dengan menyampingkan pengetahuan mengenai baris kode yang ada didalamnya. Fokus yang diuji dalam pengujian *black box* adalah

*input* dan *output* pada aplikasi yang dikembangkan. Penguji memberikan masukan dan mengamati keluaran yang dihasilkan oleh sistem. Hal ini memungkinkan untuk mengidentifikasi bagaimana sistem merespons tindakan pengguna apakah sesuai dengan apa yang diharapkan atau tidak. Pengujian *black box* sangat penting dilakukan karena berkaitan langsung dengan pengguna akhir, pengguna tidak akan peduli bagaimana sistem itu dikembangkan atau dirancang. Pengguna hanya berharap menerima respons yang sesuai dengan apa yang mereka *input* ke dalam sistem.



## **BAB III**

### **METODOLOGI PENELITIAN**

Pada penelitian ini menggunakan metode *waterfall*. Metode *waterfall* merupakan salah satu proses perancangan sistem secara terstruktur dengan pendekatan yang sistematis. Metode ini berjalan secara berurutan dengan melalui beberapa tahapan di mana luaran dari satu tahapan akan digunakan sebagai masukan di tahapan berikutnya. Berikut ini merupakan penjabaran dari tahapan pengembangan tersebut yang disesuaikan dengan penelitian.

#### **3.1 Analisis Kebutuhan**

Analisis dilakukan untuk mencari dan merumuskan apa saja kebutuhan sistem yang akan dikembangkan. Dengan dilakukannya analisis diharapkan nantinya sistem yang akan dikembangkan dapat mencapai target dan sesuai dengan apa yang diinginkan. Aplikasi sistem jemput sampah berbasis android untuk rumah kos dan sekitar area kampus merupakan sistem yang dapat menghubungkan antara pengguna yaitu masyarakat dan pengepul atau pengelola sampah dengan fitur utamanya adalah proses jemput sampah yang dilakukan oleh petugas atas permintaan dari pengguna. Sistem yang akan dikembangkan dapat melakukan permintaan penjemputan sampah dan pemrosesan penjemputan sampah. Dengan analisis yang dilakukan dapat membantu untuk menentukan kebutuhan apa saja yang berkaitan dengan sistem yang akan dikembangkan. Mulai dari kebutuhan masukan, proses-proses yang ada di dalam sistem dan juga keluaran sistem.

Terdapat tiga pengguna yang akan terlibat dalam proses penjemputan sampah yaitu masyarakat, admin dan petugas sampah. Masyarakat melakukan permintaan penjemputan sampah yang kemudian akan diproses oleh admin. Admin akan meneruskan permintaan penjemputan oleh masyarakat ke petugas untuk bisa dijemput sesuai dengan titik penjemputan. Sistem yang akan dikembangkan juga akan disesuaikan dengan jenis penggunanya. Karena setiap pengguna memiliki kebutuhan dan *use case* yang sangat berbeda, sistem yang akan dikembangkan akan dipisahkan berdasarkan jenis penggunanya. Pendekatan ini juga sudah banyak diadopsi oleh aplikasi-aplikasi yang sudah ada seperti aplikasi *e-commerce* Tokopedia yang memisahkan antara aplikasi penjual dan pembeli dan juga pada aplikasi *ride-hailing* seperti Gojek dengan memisahkan antara pengendara dan penumpang. Pendekatan ini juga berakibat pada ukuran aplikasi yang lebih kecil dan dapat meningkatkan kinerja aplikasi.

Kemudian untuk alasan keamanan, aplikasi untuk admin akan jauh lebih aman jika tidak bisa diakses oleh pengguna lain baik itu masyarakat atau maupun petugas sampah.

### 3.1.1 Analisis Kebutuhan Masukan

Dalam aplikasi yang akan dikembangkan akan ada beberapa masukan berdasarkan penggunaannya yaitu masyarakat, admin dan petugas yang melakukan penjemputan sampah.

#### a. Kebutuhan masukan oleh masyarakat:

1. Nomor telepon, nama dan kata sandi untuk pendaftaran dan masuk ke dalam sistem.
2. Alamat, diisi dengan alamat lokasi penjemputan pengguna.
3. Informasi sampah yang akan dikirimkan ke sistem.

#### b. Kebutuhan masukan oleh admin:

1. Nomor telepon dan kata sandi untuk masuk ke dalam sistem.
2. Lokasi, diisi dengan lokasi tempat pengepul sampah.
3. Informasi petugas seperti nama, nomor telepon dan kata sandi untuk register petugas baru.

#### c. Kebutuhan masukan oleh petugas:

1. Nomor telepon dan kata sandi untuk masuk ke dalam sistem.

### 3.1.2 Analisis Kebutuhan Proses

Kebutuhan proses yang dapat dilakukan oleh pengguna aplikasi jemput sampah berbasis android ini adalah:

#### a. Kebutuhan proses pada pengguna sebagai masyarakat:

1. Proses pendaftaran akun baru.
2. Proses login.
3. Proses permintaan jemput sampah.
4. Proses edit data *username* dan *password*.
5. Proses edit data alamat.
6. Proses batalkan permintaan jemput sampah.

#### b. Kebutuhan proses pada pengguna sebagai administrator:

1. Proses login.
2. Proses kelola data sampah (tambah data, hapus data dan edit data).
3. Proses kelola data petugas (tambah data, hapus data dan edit data).
4. Proses konfirmasi permintaan penjemputan sampah oleh masyarakat.

5. Proses pembatalan permintaan penjemputan sampah.
- c. Kebutuhan proses pada pengguna sebagai petugas:
1. Proses penjemputan sampah.
  2. Proses konfirmasi penjemputan sampah.

### 3.1.3 Analisis Kebutuhan Keluaran

Kebutuhan keluaran yang ada pada sistem berdasarkan penggunanya adalah:

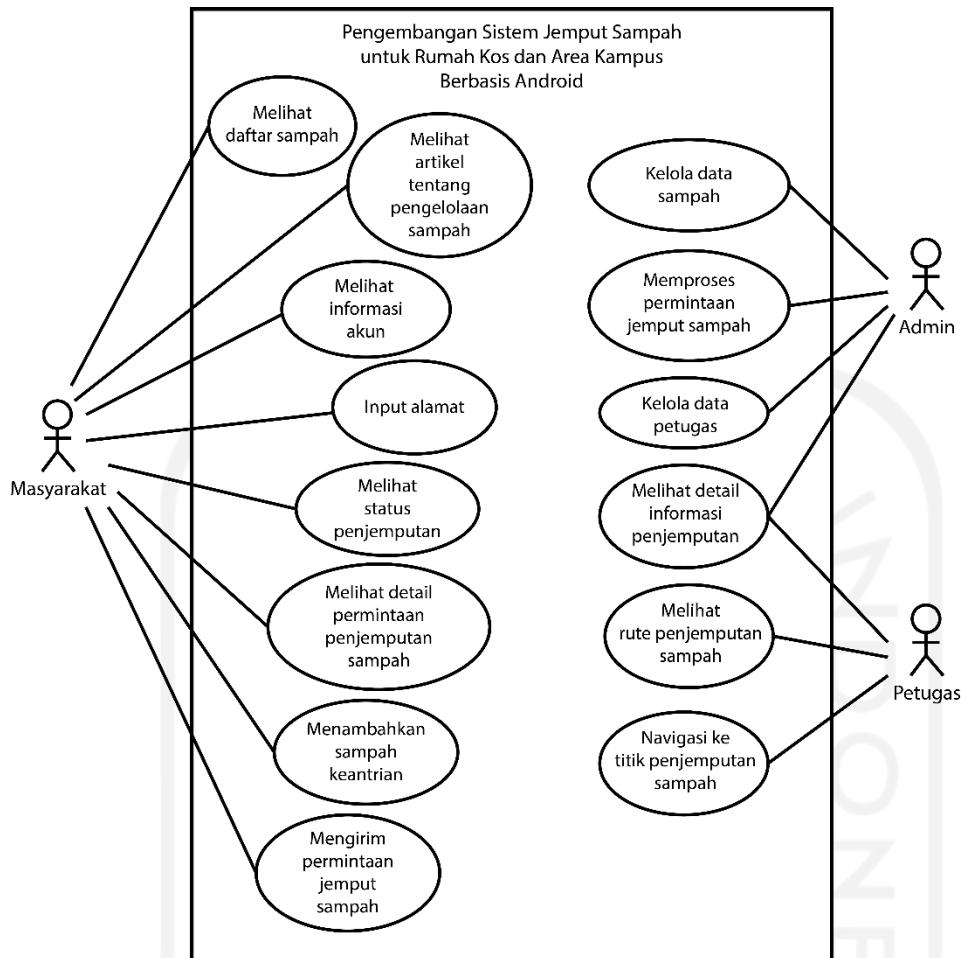
- a. Kebutuhan keluaran pada pengguna sebagai masyarakat:
  1. Informasi penjemputan sampah telah terkirim.
  2. Informasi status penjemputan sampah.
  3. Informasi saldo dan poin
  4. Informasi penjemputan berhasil.
  5. Informasi penjemputan dibatalkan.
- b. Kebutuhan keluaran pada pengguna sebagai administrator:
  1. Informasi lokasi penjemputan sampah.
  2. Informasi petugas yang tersedia.
  3. Informasi detail permintaan penjemputan sampah yang masuk.
  4. Informasi penjemputan selesai.
  5. Informasi penjemputan gagal.
- c. Kebutuhan keluaran pada pengguna sebagai petugas:
  1. Informasi lokasi penjemputan sampah.
  2. Informasi detail permintaan penjemputan sampah.
  3. Informasi navigasi menuju lokasi penjemputan sampah.

## 3.2 Perancangan Sistem

### 3.2.1 Use Case Diagram

*Use case diagram* digunakan untuk memberikan gambaran interaksi hubungan antara pengguna atau aktor dan fungsionalitas apa saja yang ada di dalam sistem yang akan dikembangkan. *Use case diagram* juga digunakan untuk mengetahui skenario bagaimana sistem bekerja. Pada sistem yang akan dikembangkan *use case diagram* menggambarkan beberapa aktor yaitu masyarakat sebagai *end-user*, admin dan juga petugas yang terhubung dengan beberapa proses yang ada di dalam sistem.





Gambar 3.1 Use case diagram

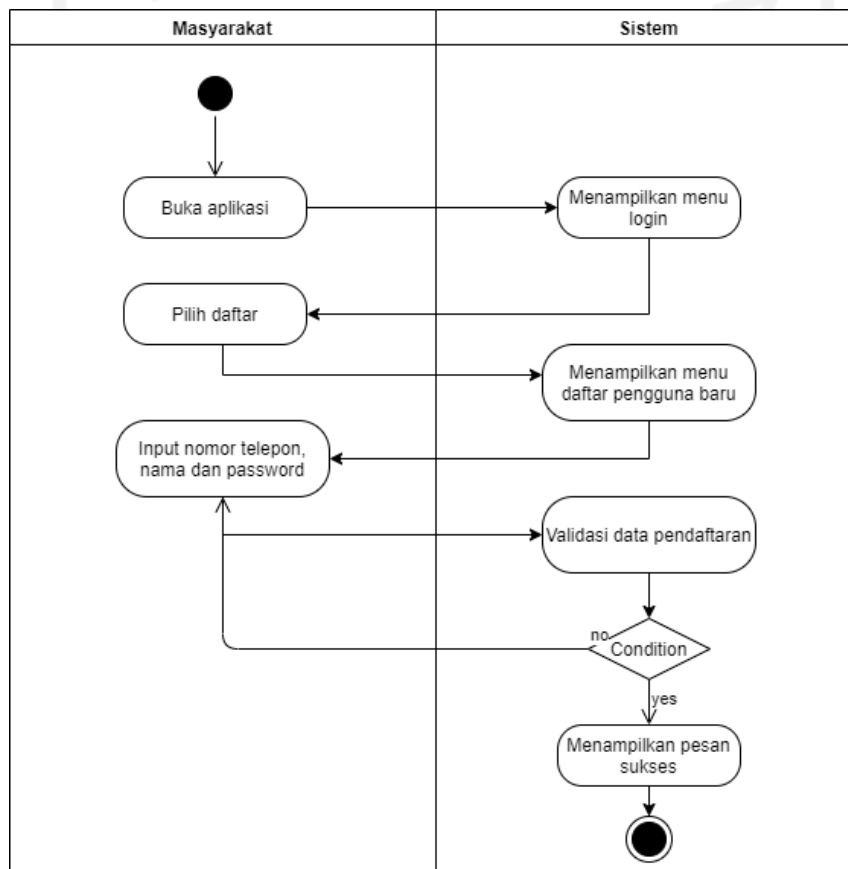
Gambar 3.1 diatas dapat dilihat bahwa pada sistem yang akan dikembangkan terdapat 3 aktor utama yaitu masyarakat, admin dan juga petugas. Pada aktor masyarakat terdapat 8 fungsionalitas yaitu melihat daftar sampah, menambahkan sampah ke antrian, mengirim permintaan penjemputan sampah, melihat artikel tentang pengelolaan sampah, melihat informasi akun, input alamat, melihat status penjemputan dan melihat detail permintaan penjemputan sampah. Kemudian untuk aktor admin memiliki 4 fungsionalitas yaitu kelola data sampah, memproses permintaan penjemputan sampah, kelola data petugas dan melihat detail penjemputan sampah. Aktor yang terakhir yaitu petugas memiliki 3 fungsionalitas yaitu melihat detail permintaan jemput sampah, melihat rute penjemputan sampah dan navigasi ke lokasi penjemputan sampah. Jadi total akan ada 15 fungsionalitas yang akan ada pada sistem yang akan dikembangkan.

### 3.2.2 Activity Diagram

*Activity Diagram* merupakan diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem dari proses status awal sampai status akhir. *Activity Diagram* juga merupakan pengembangan dari *use case* yang memiliki aktivitas. *Activity Diagram* digunakan untuk menjelaskan runtutan aktivitas yang berada disuatu proses. Berikut merupakan *Activity Diagram* yang ada dalam aplikasi yang akan dikembangkan.

#### Activity Diagram Daftar Akun Baru

Gambar 3.2 menunjukkan alur dari rangkaian aktivitas pada proses pendaftaran akun baru untuk masyarakat.

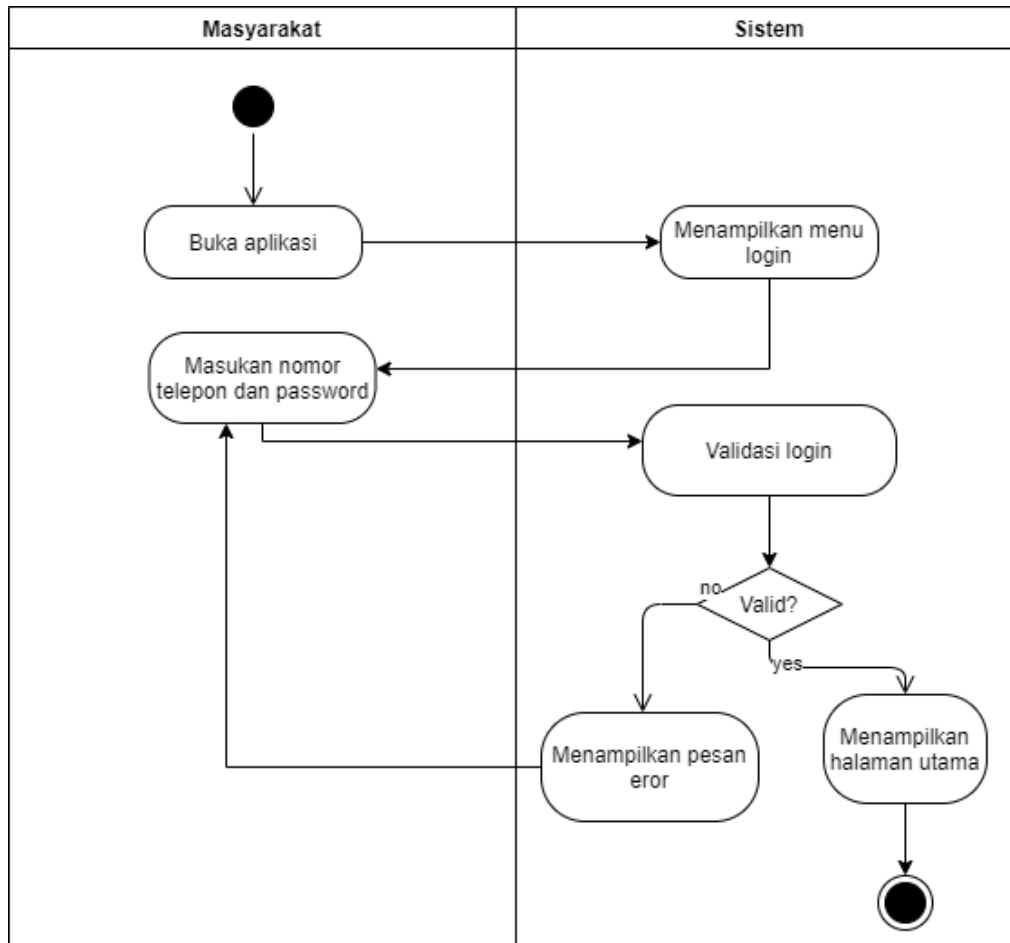


Gambar 3.2 Activity Diagram daftar akun baru.

Untuk melakukan pendaftaran akun baru, pengguna sebagai masyarakat cukup membuka aplikasi kemudian sistem akan menampilkan halaman login. Kemudian menekan tombol daftar, sistem akan menampilkan menu daftar akun. Pengguna diminta untuk memasukan data nomor telepon, nama dan password. Sistem akan memvalidasi apakah akun sudah ada atau belum, jika sudah akan menampilkan pesan eror dan jika berhasil sistem akan menampilkan pesan sukses.

### Activity Diagram Login

Gambar 3.3 menunjukkan alur dari rangkaian aktivitas pada proses login untuk masyarakat, admin dan juga petugas.

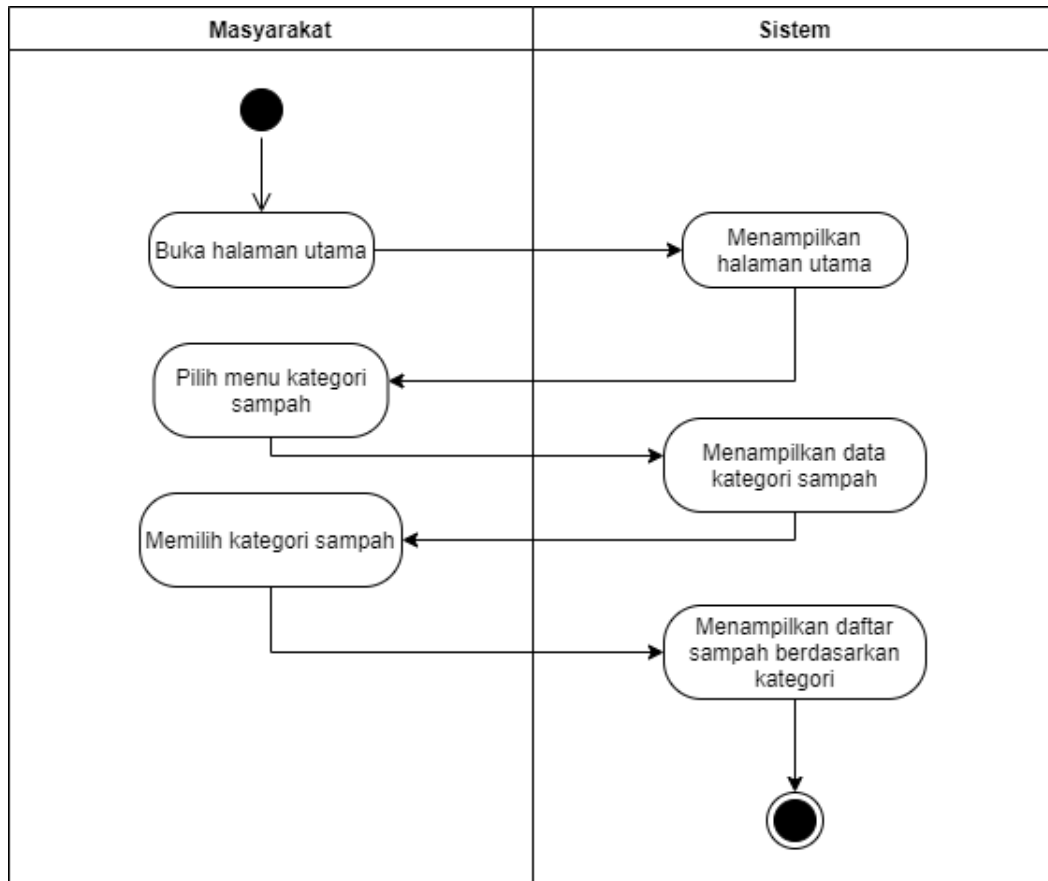


Gambar 3.3 Activity Diagram login.

Setelah membuka aplikasi pengguna akan diarahkan pada menu login. Untuk melakukan login ke dalam aplikasi pengguna diminta untuk memasukkan data berupa nomor telepon dan password yang sudah terdaftar pada sistem. Untuk memastikan bahwa pengguna sudah terdaftar sistem akan melakukan validasi apakah pengguna sudah terdaftar. Jika belum terdaftar maka sistem akan menampilkan pesan eror dan jika berhasil maka pengguna akan diarahkan ke menu utama aplikasi.

### Activity Diagram Melihat Daftar Sampah

Gambar 3.4 menunjukkan alur dari rangkaian aktivitas pada proses melihat daftar sampah yang tersedia di dalam kategori.

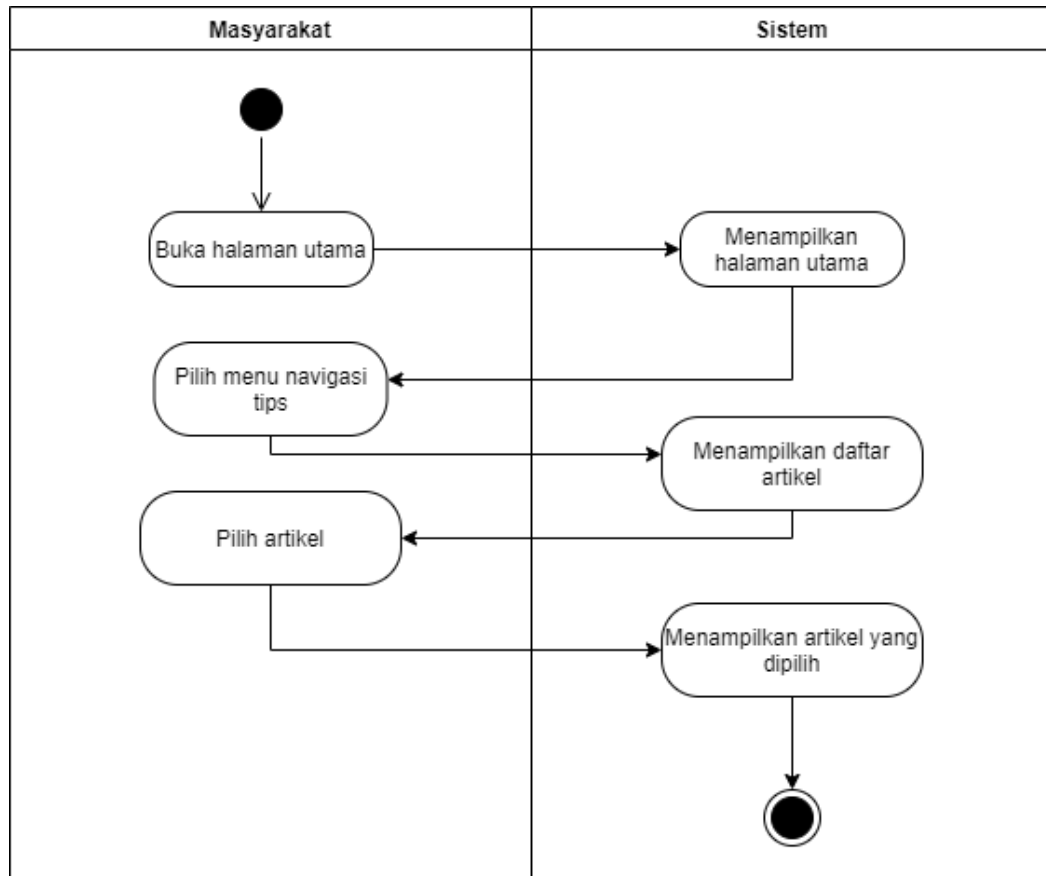


Gambar 3.4 Activity Diagram melihat daftar sampah.

Pada menu utama aplikasi terdapat menu kategori sampah, menu kategori digunakan untuk menampilkan data kategori sampah. Dari daftar kategori yang ditampilkan masyarakat bisa memilih kategori yang tersedia untuk menampilkan daftar sampah berdasarkan kategori.

### Activity Diagram Melihat Artikel

Gambar 3.5 menunjukkan alur dari rangkaian aktivitas pada proses melihat artikel tentang pengelolaan sampah.

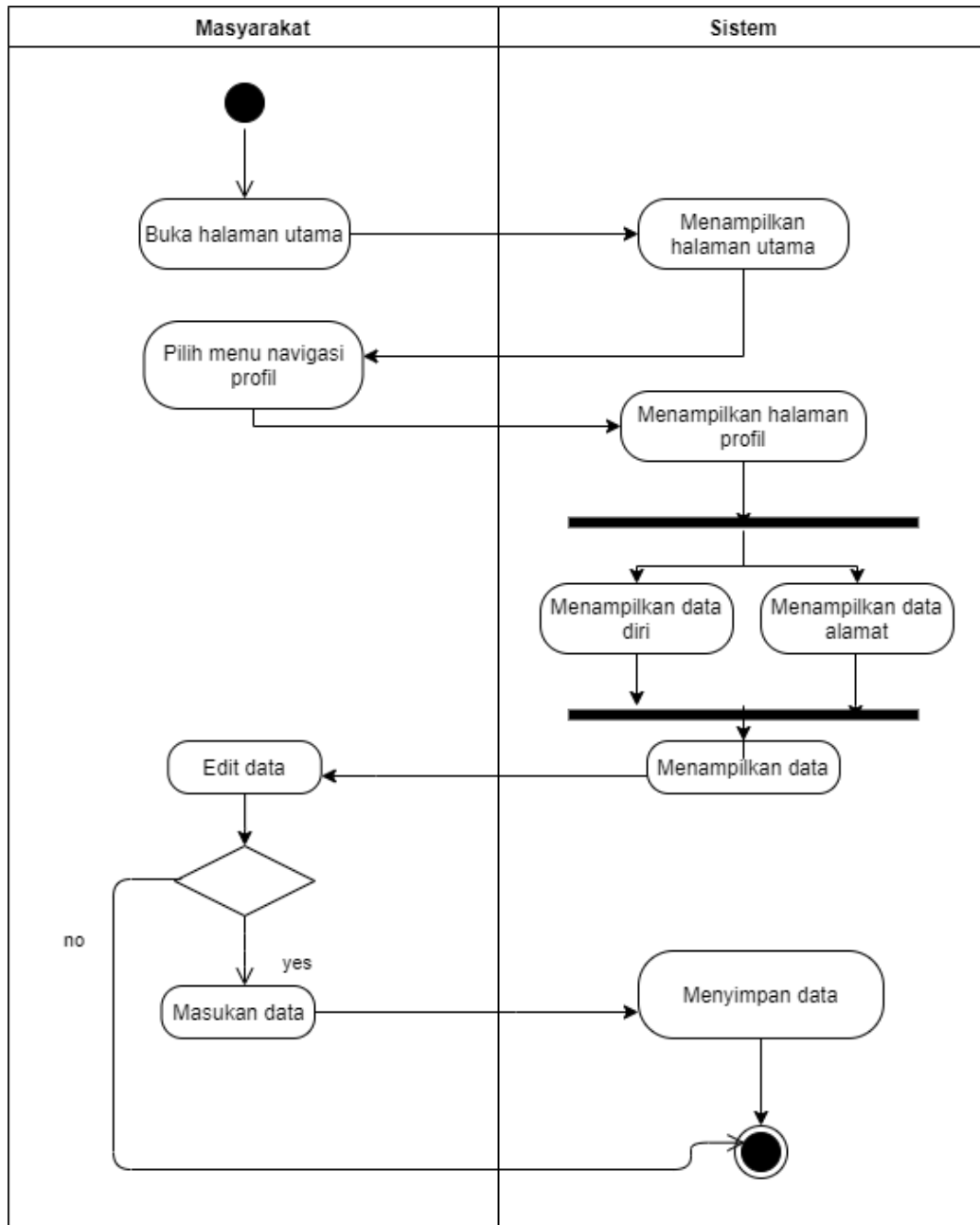


Gambar 3.5 Activity Diagram melihat artikel.

Untuk menampilkan daftar artikel tentang pengelolaan sampah masyarakat cukup memilih menu tips pada halaman utama. Sistem akan menampilkan daftar artikel yang diminta oleh masyarakat.

### Activity Diagram Melihat Informasi Akun

Gambar 3.6 menunjukkan alur dari rangkaian aktivitas pada proses melihat informasi akun untuk masyarakat.

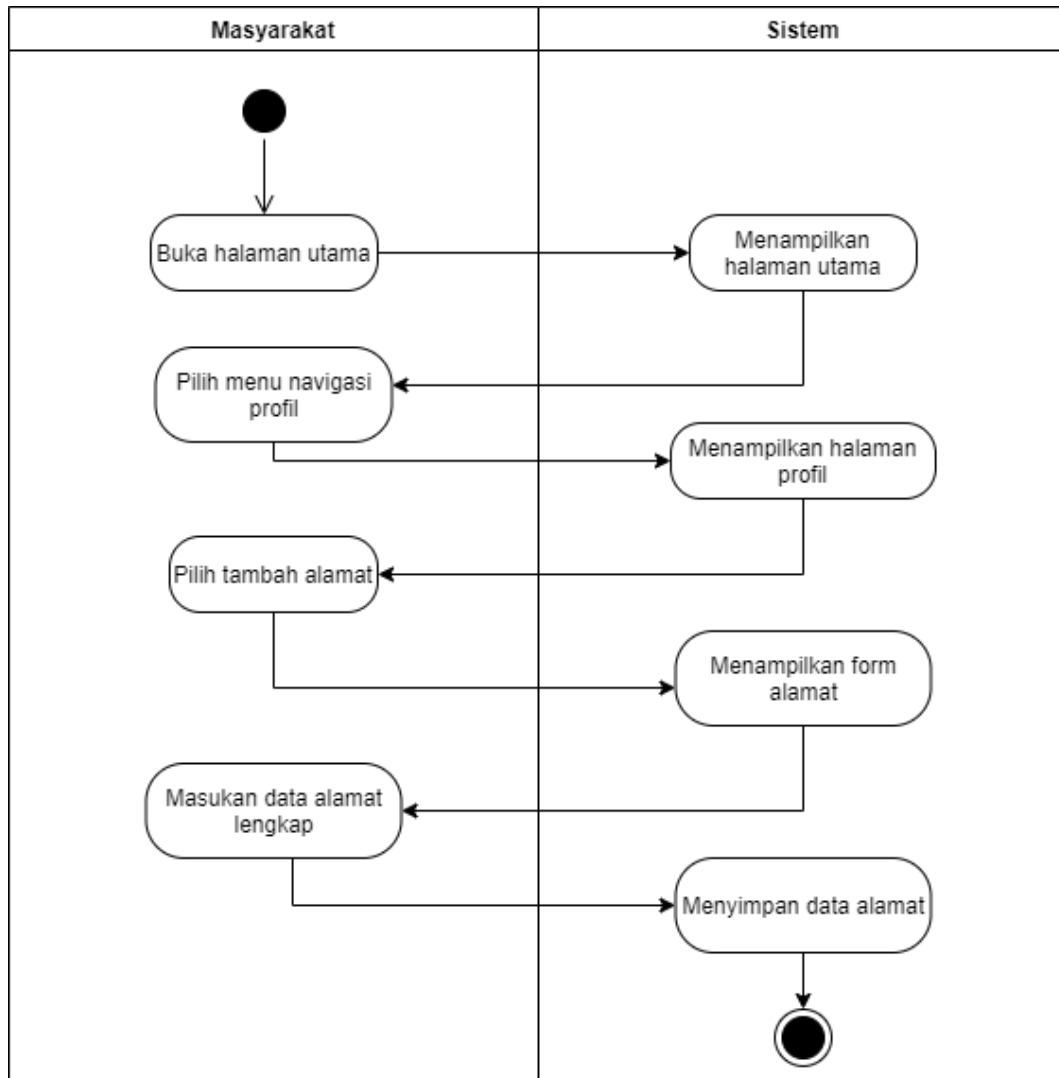


Gambar 3.6 Activity Diagram melihat informasi akun.

Ketika masyarakat memilih menu profil, maka sistem akan menampilkan informasi akun berupa informasi data diri dan juga data alamat. Jika masyarakat ingin mengubah data maka dapat mengisi *forms* untuk data yang ingin diubah, setelah data diisi maka sistem akan menyimpan data ke database sistem.

### Activity Diagram Input Alamat

Gambar 3.7 menunjukkan alur dari rangkaian aktivitas pada proses input alamat untuk masyarakat.

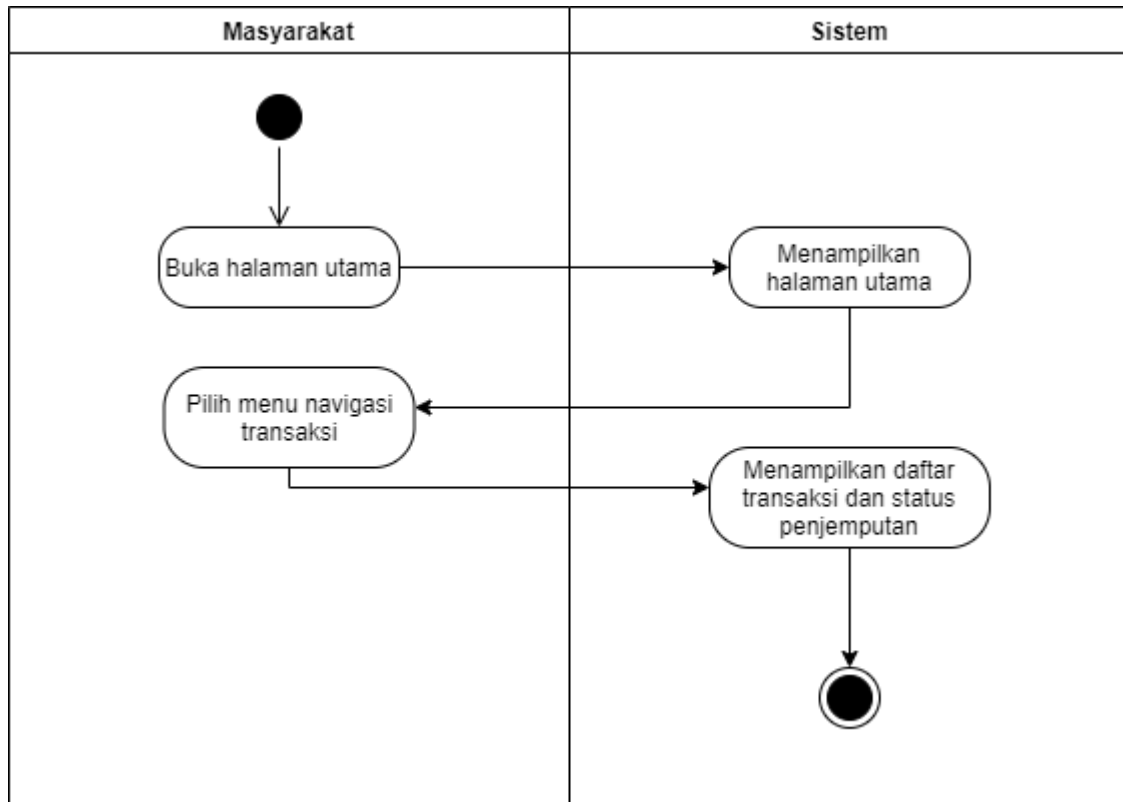


Gambar 3.7 Activity Diagram input alamat.

Untuk menambahkan alamat baru, masyarakat memilih pada menu profil. Sistem akan menampilkan halaman profil, selanjutnya pilih pada bagian tambah alamat. Masyarakat akan diminta untuk mengisi alamat pada *forms* yang ditampilkan. Setelah alamat sudah diisi dengan benar sistem akan menyimpan alamat.

### Activity Diagram Melihat Status Penjemputan

Gambar 3.8 menunjukkan alur dari rangkaian aktivitas pada proses melihat status penjemputan.



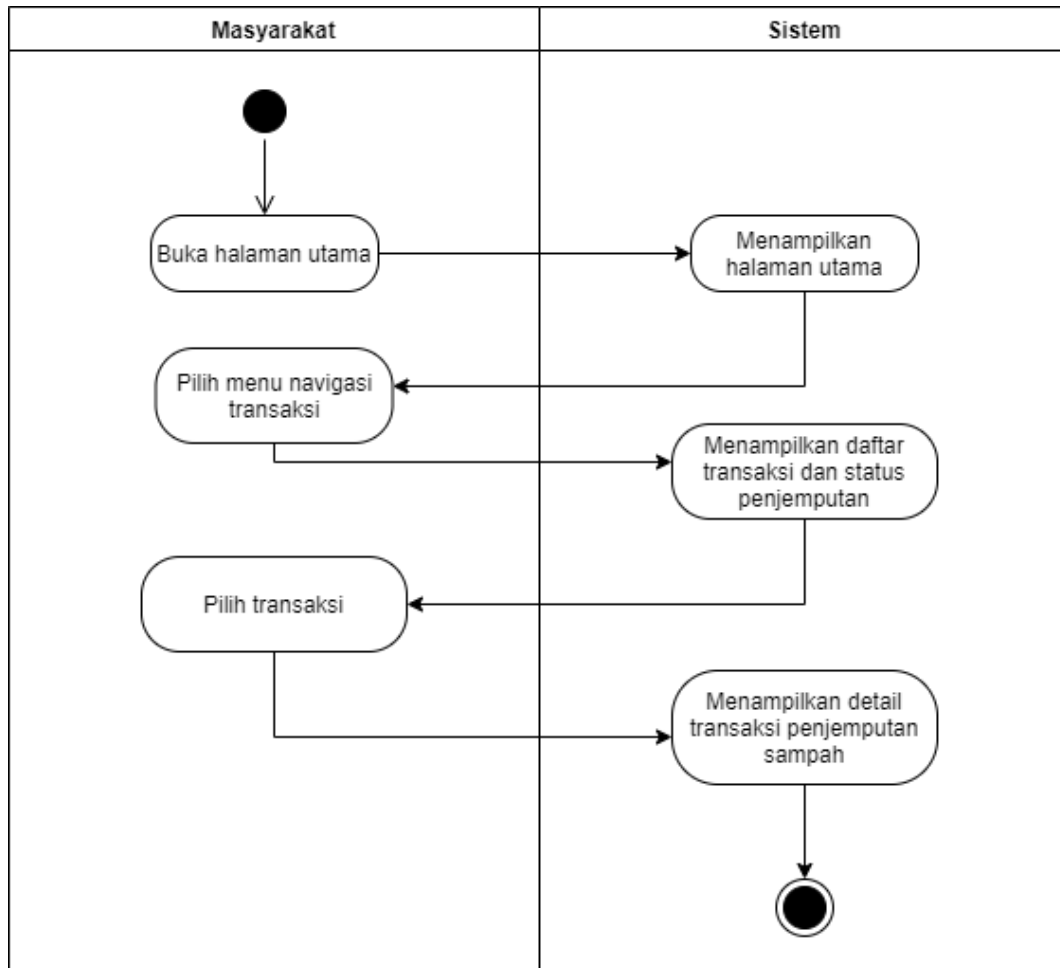
Gambar 3.8 Activity Diagram melihat status penjemputan

Untuk melihat status penjemputan masyarakat memilih menu transaksi pada halaman utama. Kemudian sistem akan menampilkan daftar transaksi yang sedang berlangsung beserta status penjemputannya. Status penjemputan ada 4 yaitu status belum diproses, status berhasil, status ditolak dan yang terakhir status dalam penjemputan.



### Activity Diagram Melihat Detail Penjemputan Sampah

Gambar 3.9 menunjukkan alur dari rangkaian aktivitas pada proses melihat detail penjemputan sampah.

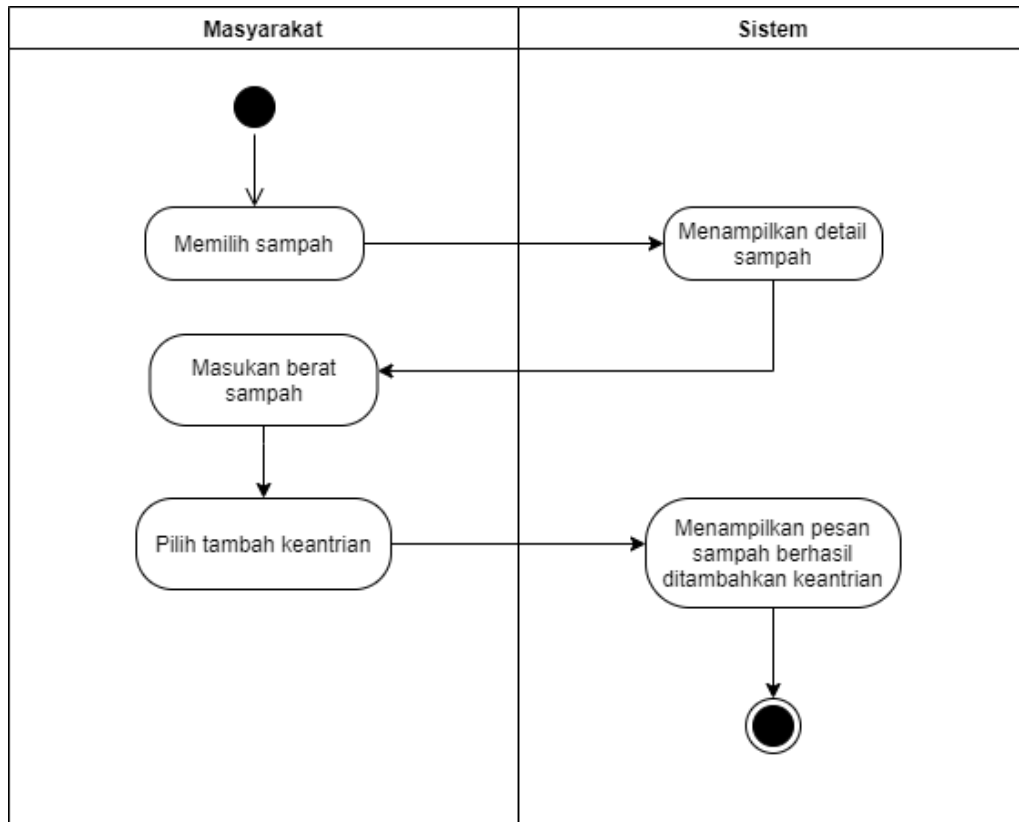


Gambar 3.9 Activity Diagram melihat detail penjemputan sampah

Untuk melihat detail transaksi penjemputan sampah, masyarakat memilih menu transaksi pada halaman utama kemudian memilih transaksi penjemputan sampah mana yang ingin dilihat secara detail. Detail transaksi berisi informasi status penjemputan, informasi petugas, informasi akun dan informasi penjemputan.

### Activity Diagram Menambahkan Sampah Keantrian

Gambar 3.10 menunjukkan alur dari rangkaian aktivitas pada proses menambahkan sampah keantrian.

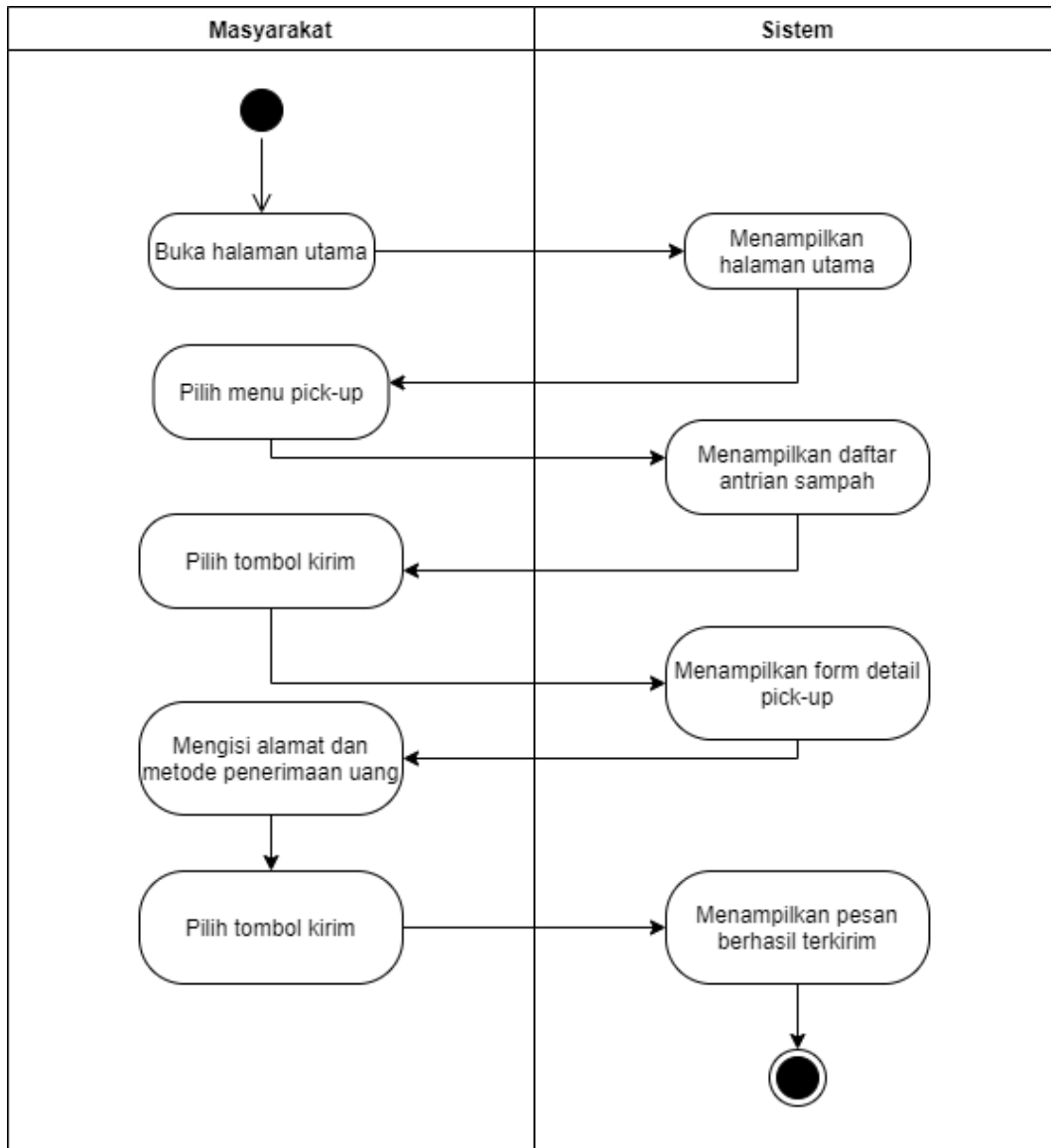


Gambar 3.10 Activity Diagram menambahkan sampah keantrian.

Untuk menambahkan sampah ke antrian masyarakat memilih sampah yang ingin ditambahkan, kemudian sistem akan menampilkan detail sampah. Masyarakat diminta untuk memasukan data berat sampah yang ingin ditambahkan ke antrian. Setelah memasukan berat sampah kemudian pengguna memilih tombol tambah ke antrian, maka sistem akan menampilkan pesan sampah berhasil ditambahkan.

### Activity Diagram Mengirim Permintaan Jemput Sampah

Gambar 3.11 menunjukkan alur dari rangkaian aktivitas pada proses mengirim permintaan jemput sampah.

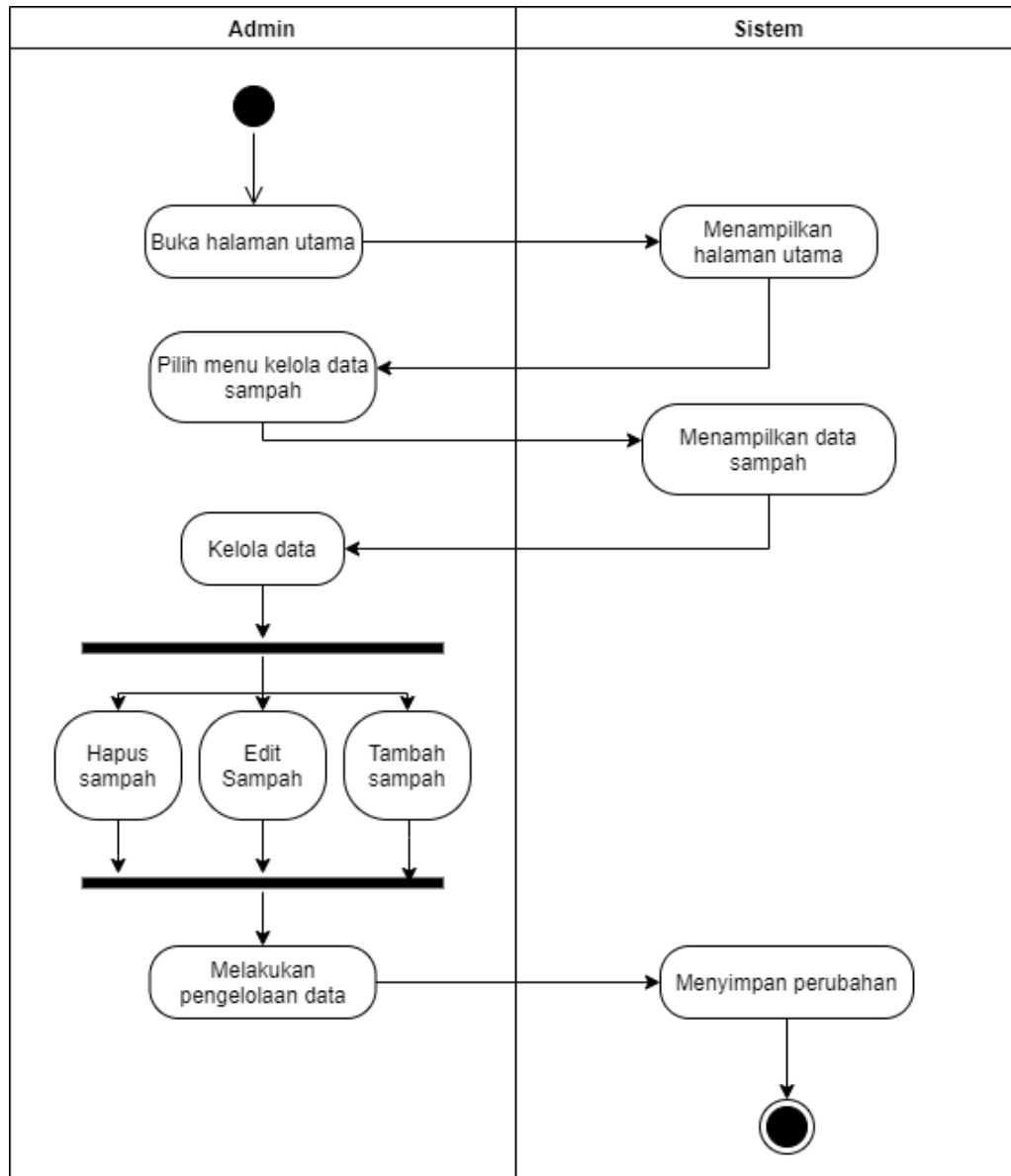


Gambar 3.11 Activity Diagram mengirim permintaan jemput sampah.

Untuk mengirim permintaan jemput sampah, masyarakat memilih menu menunggu *pick-up* pada halaman utama. Sistem akan menampilkan daftar antrean sampah. Kemudian masyarakat akan diminta untuk mengisi data detail penjemputan berupa alamat dan penerimaan uang. Untuk mengirim masyarakat cukup memilih tombol kirim, maka sistem akan menampilkan pesan berhasil terkirim.

### Activity Diagram Kelola Data Sampah

Gambar 3.12 menunjukkan alur dari rangkaian aktivitas pada proses mengirim permintaan jemput sampah.

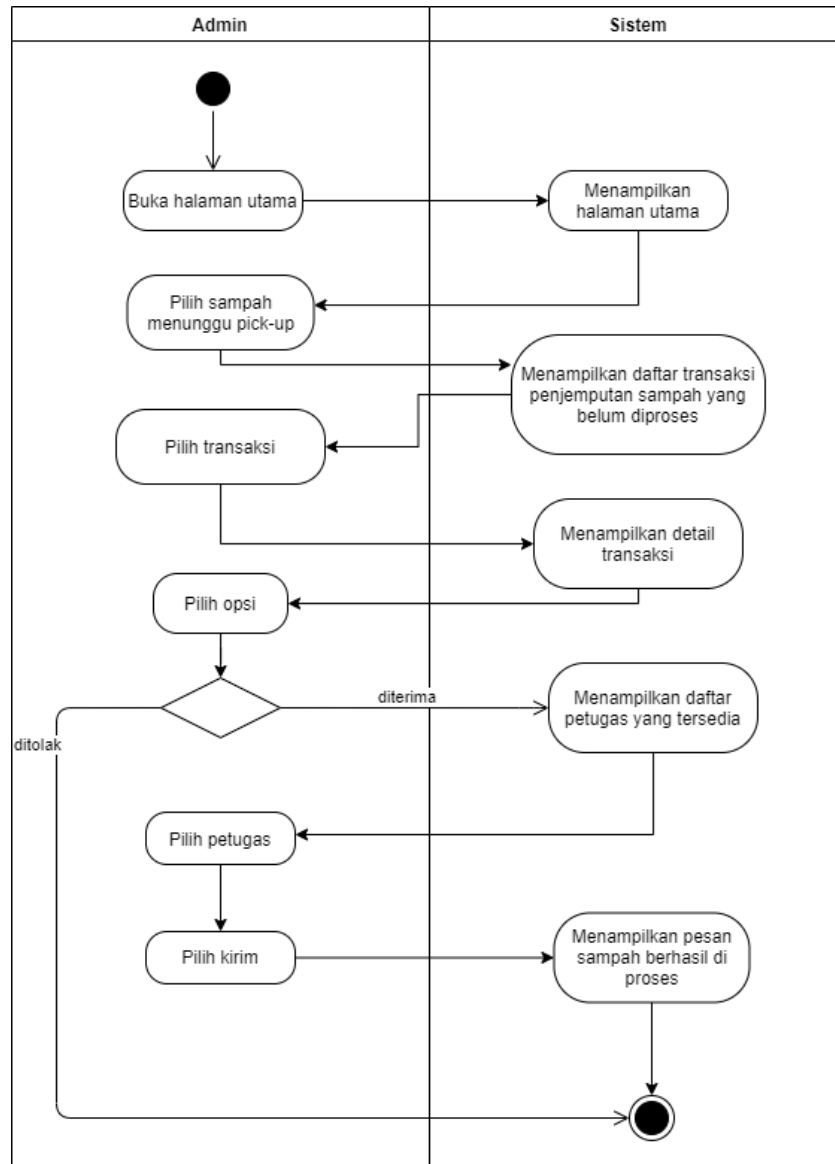


Gambar 3.12 Activity Diagram kelola data sampah.

Untuk melakukan pengelolaan data pada sampah pilih pada menu kelola data sampah di halaman utama. Kemudian sistem akan menampilkan daftar sampah yang sudah ada. Admin bisa melakukan pengelolaan data seperti menghapus sampah, edit sampah dan juga tambah sampah. Setelah melakukan pengelolaan data sampah sistem akan menyimpan perubahan yang telah dilakukan baik itu menghapus, edit ataupun menambahkan data sampah.

### Activity Diagram Memproses Permintaan Jemput Sampah

Gambar 3.13 menunjukkan alur dari rangkaian aktivitas pada proses admin memproses permintaan jemput sampah

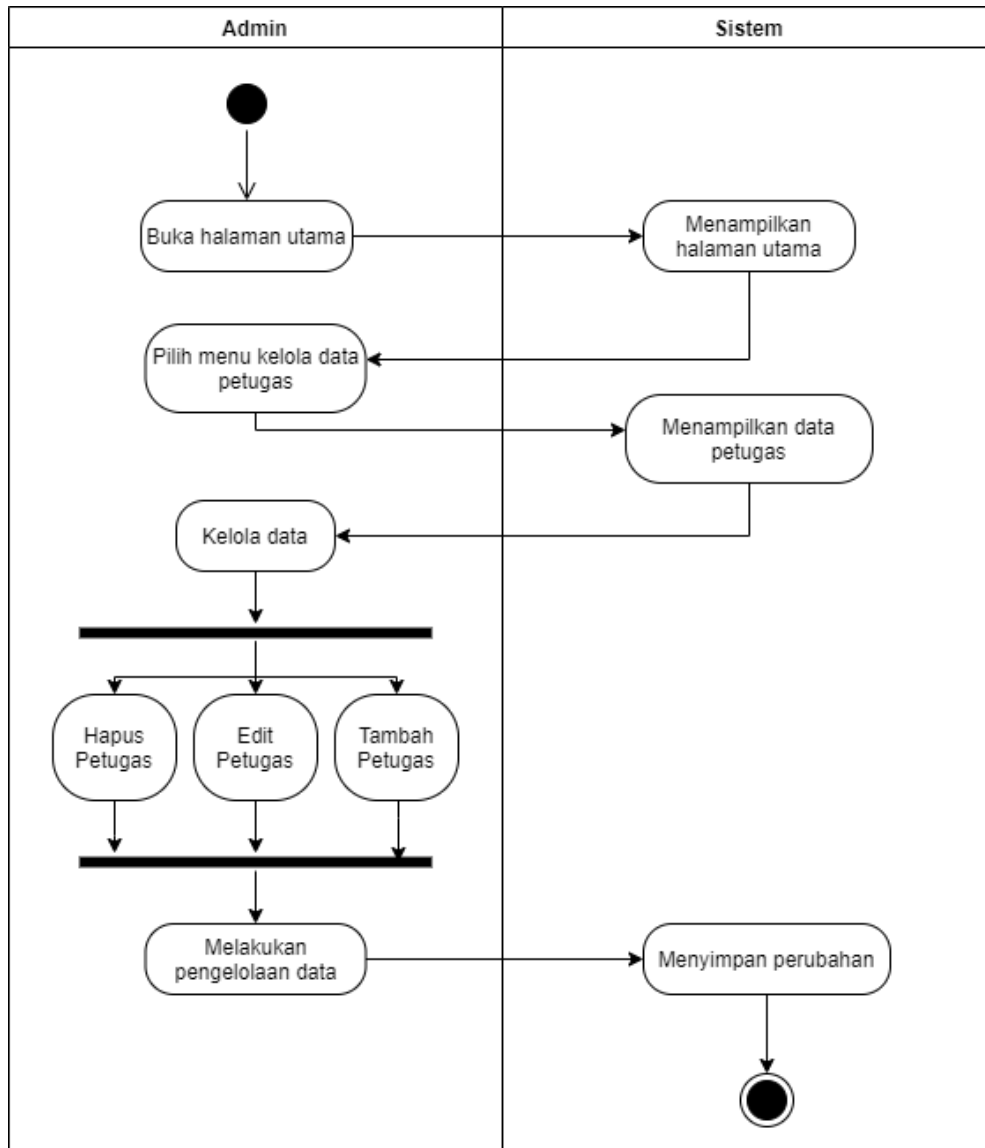


Gambar 3.13 Activity Diagram memproses permintaan jemput sampah.

Untuk memproses permintaan penjemputan pengguna sebagai admin memilih menu sampah menunggu *pick-up* pada halaman utama. Kemudian sistem akan menampilkan daftar transaksi yang belum diproses. Pilih pada transaksi yang ingin diproses kemudian pilih opsi apakah diterima atau ditolak. Jika diterima akan muncul menu daftar petugas, pilih petugas mana yang akan diberi tugas untuk menjemput sampah. Maka sistem akan menampilkan pesan berhasil dan status penjemputan akan berubah menjadi sedang dalam penjemputan. Jika ditolak maka sistem akan mengirimkan notifikasi ke masyarakat bahwa permintaan penjemputan ditolak.

### Activity Diagram Kelola Data Petugas

Gambar 3.14 menunjukkan alur dari rangkaian aktivitas pada proses admin melakukan pengelolaan data petugas.

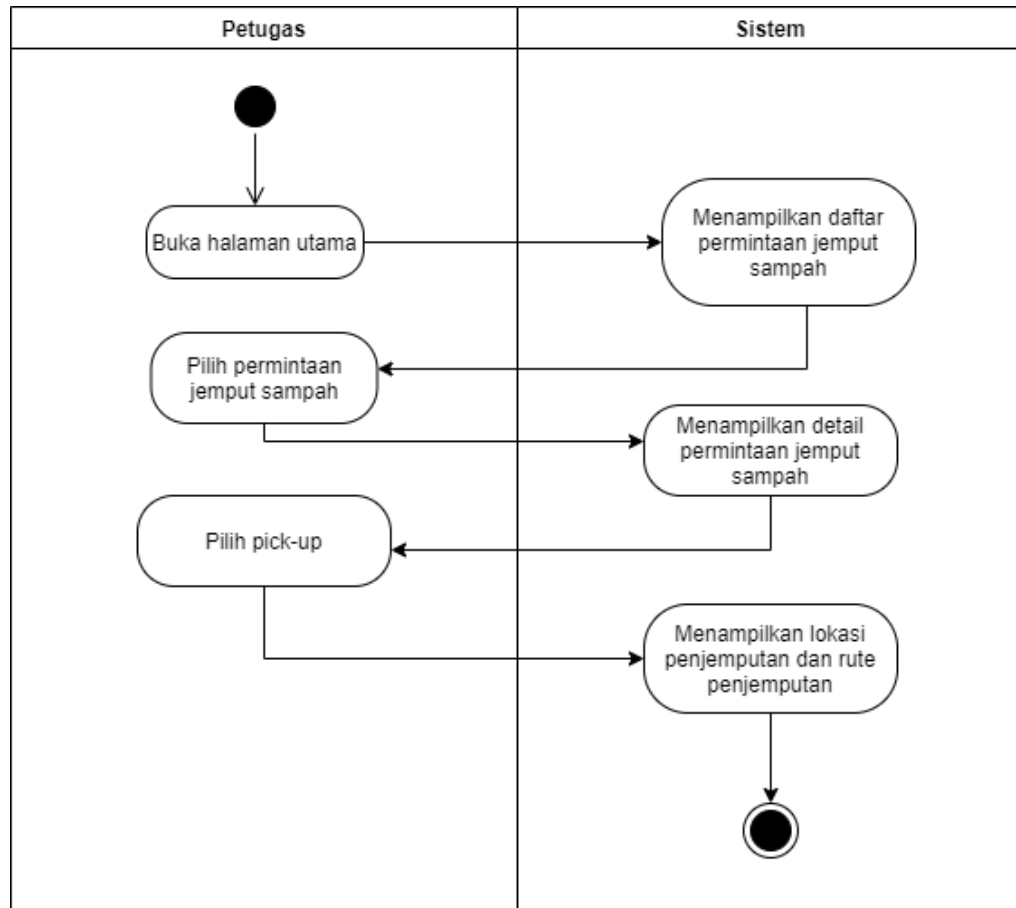


Gambar 3.14 Activity Diagram kelola data petugas

Untuk mengelola data petugas, admin memilih menu kelola data petugas pada halaman utama. Sistem akan menampilkan daftar petugas yang aktif atau beroperasi. Admin bisa mengelola data petugas seperti menambah petugas, menghapus petugas dan mengedit data petugas.

### Activity Diagram Melihat Rute Penjemputan Sampah

Gambar 3.15 menunjukkan alur dari rangkaian aktivitas pada petugas melihat rute penjemputan sampah.

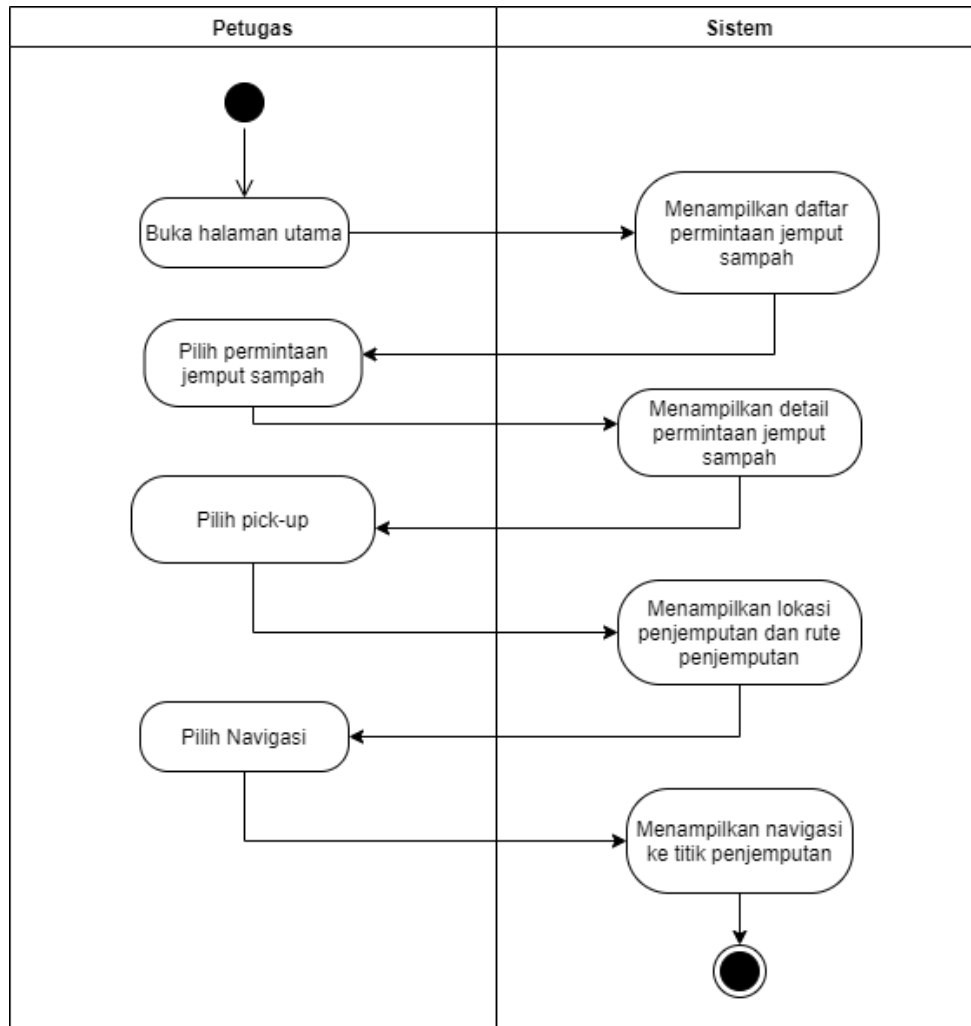


Gambar 3.15 Activity Diagram memproses melihat rute penjemputan sampah.

Untuk menampilkan rute penjemputan sampah pada halaman utama akan ditampilkan daftar permintaan jemput sampah yang sudah diproses admin. Petugas memilih permintaan jemput sampah mana yang ingin dilihat rutenya. Sistem akan menampilkan detail permintaan jemput sampah. Kemudian petugas pilih tombol *pick-up*, sistem akan menampilkan lokasi penjemputan dan rute penjemputan sampah dari lokasi petugas berada ke titik lokasi penjemputan.

### Activity Diagram Navigasi ke Titik Penjemputan Sampah

Gambar 3.16 menunjukkan alur dari rangkaian aktivitas pada proses navigasi ke titik penjemputan sampah.



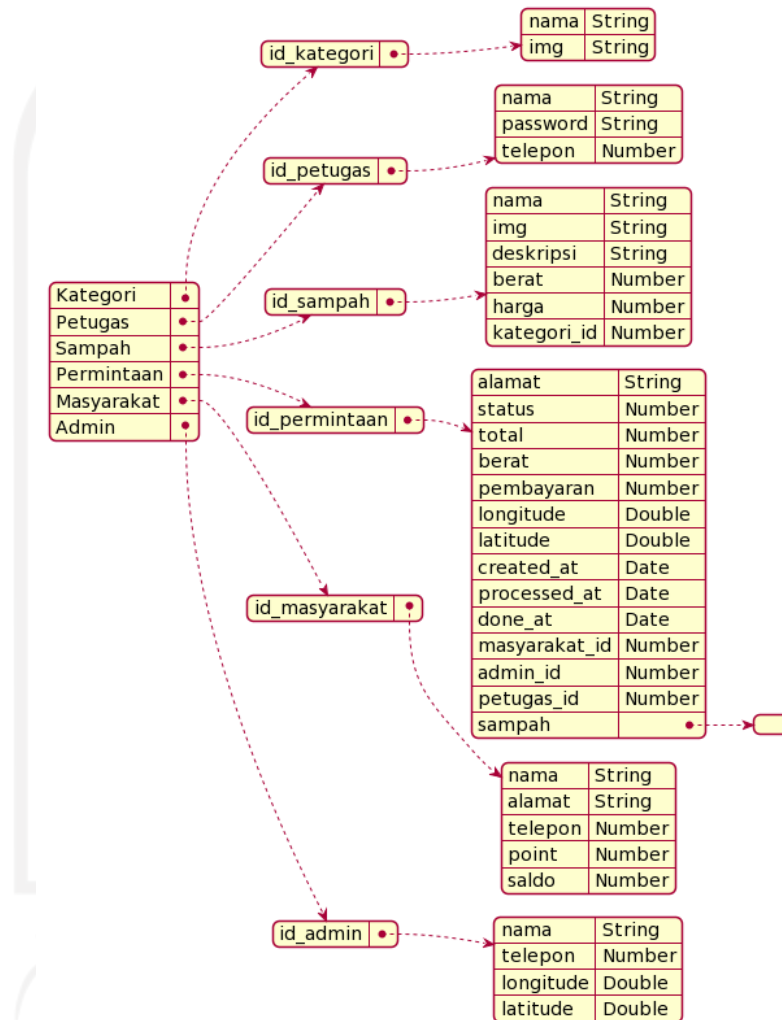
Gambar 3.16 Activity Diagram navigasi ke titik penjemputan sampah.

Pada halaman utama pilih salah satu permintaan jemput sampah, kemudian akan ditampilkan detail permintaan jemput sampah. Pilih *pick-up*, sistem akan menampilkan rute penjemputan. Kemudian pilih tombol navigasi, sistem akan menampilkan navigasi dari lokasi petugas berada menuju titik penjemputan sampah.



### 3.2.3 Perancangan Database

Database yang digunakan pada pengembangan aplikasi jemput sampah ini menggunakan *Firestore Realtime Database* yang struktur datanya berupa JSON. Terdapat 6 objek yang akan ada pada rancangan database yaitu, Masyarakat, Petugas, Admin, Permintaan, Sampah dan Kategori. Gambar 3.17 menunjukkan gambaran skema data JSON yang telah dirancang yang kemudian akan diimplementasikan pada sistem dengan *Firestore Realtime Database*.



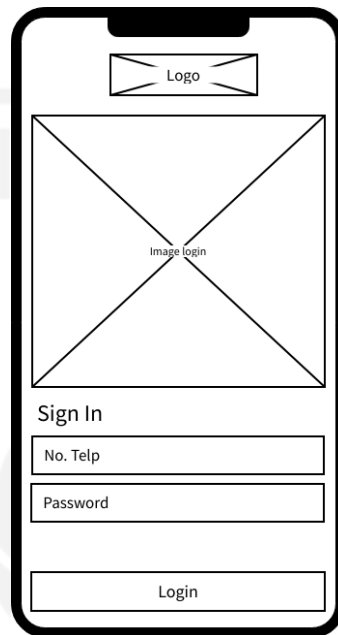
Gambar 3.17 Skema data JSON

### 3.2.4 Perancangan Antarmuka

Perancangan desain antarmuka digunakan untuk membuat gambaran *layout*, komponen apa saja yang akan digunakan dan juga untuk mendefinisikan fungsionalitas apa saja yang akan ada pada aplikasi dalam bentuk visual. Tahapan ini sangat penting, karena berhubungan langsung dengan *end-user*. Berikut merupakan perancangan antarmuka pada aplikasi yang akan dikembangkan.

### Halaman Login

Halaman login menampilkan formulir berisi nomor telepon dan juga *password* yang digunakan untuk pengguna melakukan login ke dalam aplikasi. Halaman login digunakan untuk ketiga aktor yaitu masyarakat, admin dan juga petugas. Gambar 3.18 menunjukkan rancangan antarmuka halaman login.



Gambar 3.18 Rancangan halaman login.

### Halaman Daftar Akun

Pada halaman daftar akun masyarakat diminta untuk memasukkan data nomor telepon, nama dan password. Data-data yang dimasukan digunakan untuk mendaftar sebagai pengguna baru. Setiap data yang dimasukan akan dicek apakah sudah terdaftar atau belum. Gambar 3.19 menunjukkan rancangan antarmuka halaman daftar akun.

Sign Up

No. Telp

Nama

Password

Daftar

Gambar 3.19 Rancangan halaman daftar akun.

### Halaman Utama Aplikasi untuk Masyarakat

Pada halaman utama aplikasi untuk masyarakat akan menampilkan beberapa informasi kepada masyarakat. Diantaranya informasi akun berupa nama, saldo dan juga poin. Gambar 3.20 menunjukkan rancangan halaman utama pada masyarakat.

Logo

Illustration

Saldo Rp 10000

Point 100

Icon Label

Icon Label

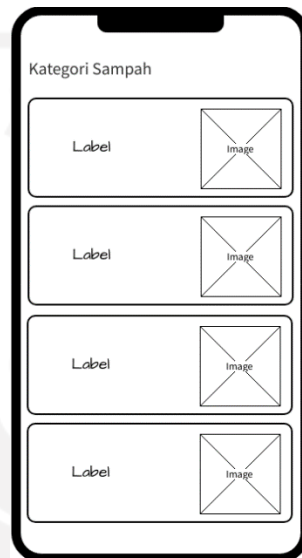
Icon Label

Icon Label

Gambar 3.20 Rancangan halaman utama untuk masyarakat.

### Halaman Kategori Sampah

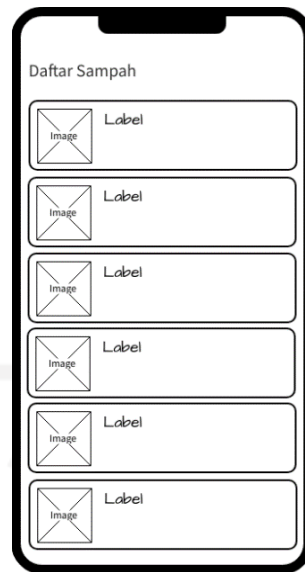
Pada halaman kategori sampah menampilkan daftar kategori sampah yang tersedia. Data yang ditampilkan berupa nama kategori dan juga gambar kategori. Gambar 3.21 menunjukkan rancangan antarmuka halaman kategori sampah.



Gambar 3.21 Rancangan halaman kategori sampah.

### Halaman Daftar Sampah

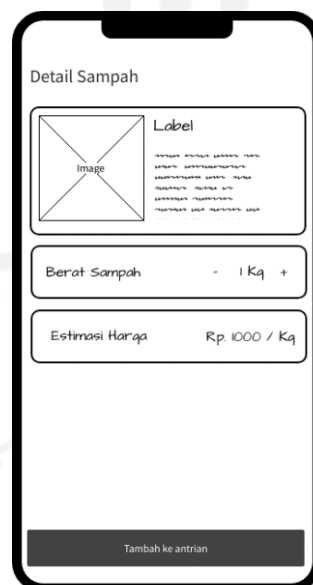
Pada halaman daftar sampah akan ditampilkan daftar data sampah berdasarkan kategori yang dipilih. Data yang ditampilkan berupa gambar sampah dan juga nama sampah. Gambar 3.22 menunjukkan rancangan antarmuka halaman daftar sampah.



Gambar 3.22 Rancangan halaman daftar sampah.

### Halaman Detail Sampah

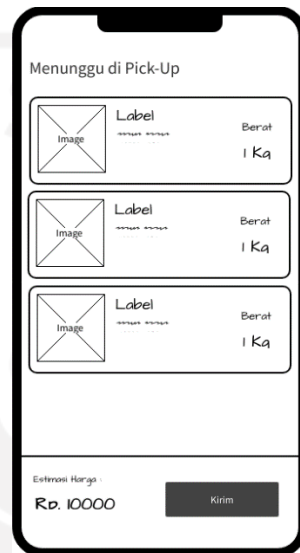
Pada halaman detail sampah ditampilkan data detail sampah yang telah dipilih masyarakat. Data sampah berupa gambar, nama, deskripsi, berat sampah dan juga estimasi harga sampah per kilogram. Gambar 3.23 menunjukkan rancangan antarmuka halaman detail sampah.



Gambar 3.23 Rancangan halaman detail sampah.

### Halaman Menunggu *Pick-up*

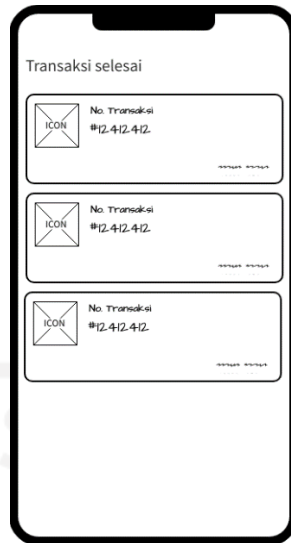
Pada halaman menunggu *pick-up* menampilkan daftar sampah yang sudah ditambahkan masyarakat ke antrean. Seluruh daftar sampah yang sudah ditambahkan akan dikalkulasikan estimasi harganya sebelum dikirim ke admin. Gambar 3.24 menunjukkan rancangan antarmuka halaman menunggu *pick-up*.



Gambar 3.24 Rancangan halaman menunggu *pick-up*.

### Halaman Transaksi Selesai

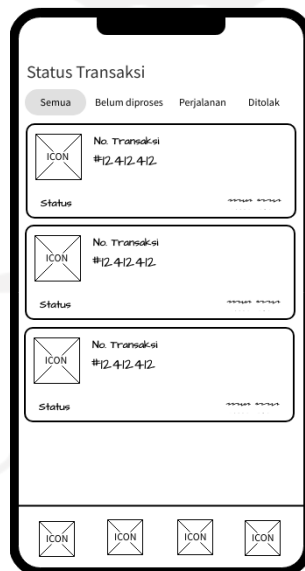
Pada halaman transaksi selesai menampilkan daftar permintaan jemput sampah yang sudah memiliki status berhasil *dipick-up* atau sudah selesai. Gambar 3.25 menunjukkan rancangan antarmuka halaman transaksi selesai.



Gambar 3.25 Rancangan halaman transaksi selesai.

### Halaman Transaksi

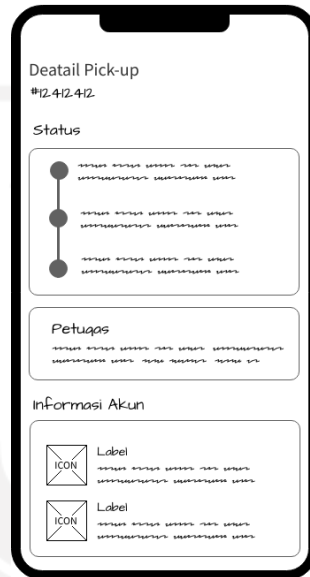
Pada halaman transaksi menampilkan daftar semua transaksi yang telah dibuat masyarakat berdasarkan status baik yang sudah selesai, ditolak, dalam perjalanan dan belum diproses. Gambar 3.26 menunjukkan rancangan antarmuka halaman transaksi.



Gambar 3.26 Rancangan halaman transaksi.

### Halaman Detail Transaksi

Pada halaman detail transaksi menampilkan detail permintaan jemput sampah yang sudah dibuat. Masyarakat bisa melacak status penjemputan dan juga mengetahui petugas yang menjemput sampah. Gambar 3.27 menunjukkan rancangan antarmuka detail transaksi.

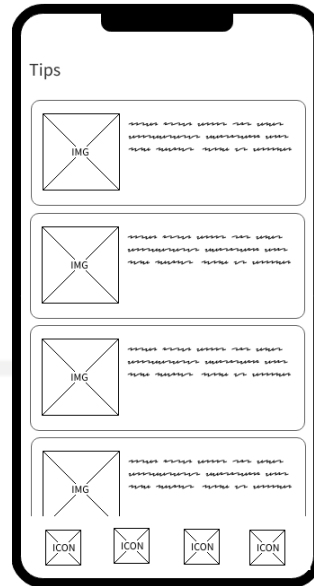


Gambar 3.27 Rancangan halaman transaksi.

### Halaman Tips

Pada halaman tips menampilkan daftar artikel pilihan mengenai tips pengelolaan sampah. Gambar 3.28 menunjukkan rancangan halaman tips.

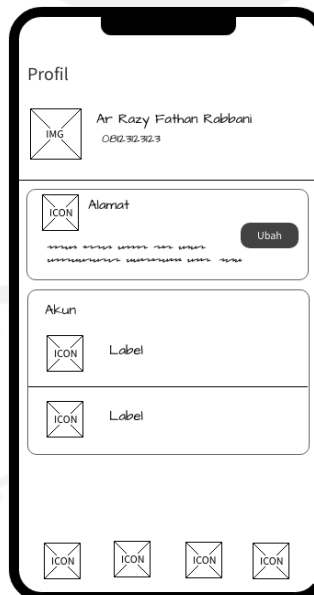




Gambar 3.28 Rancangan halaman tips.

### Halaman Profil

Pada halaman profil menampilkan data profil masyarakat berupa nama, nomor telepon dan alamat. Pada halaman profil juga terdapat menu untuk mengubah nama dan juga password. Gambar 3.29 menunjukkan rancangan antarmuka halaman profil.

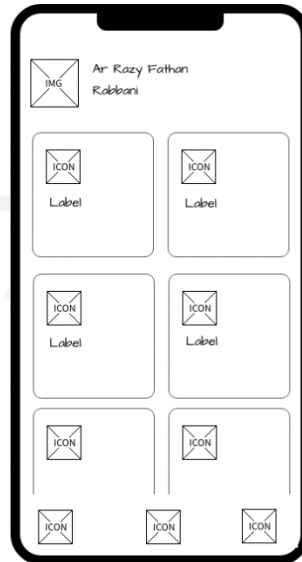


Gambar 3.29 Rancangan halaman profil.

### Halaman Utama Aplikasi untuk Admin

Pada halaman utama aplikasi untuk admin menampilkan beberapa menu untuk navigasi ke menu lain seperti menu kelola sampah, kelola petugas, sampah menunggu *pick-up*, transaksi

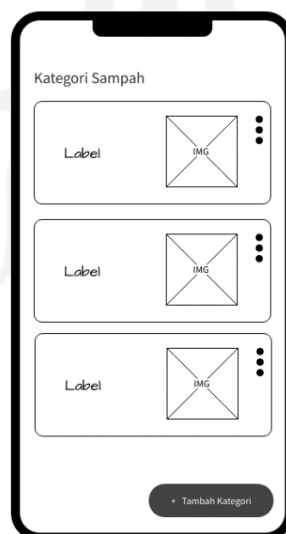
ditolak dan transaksi berhasil. Gambar 3.30 menunjukkan rancangan antarmuka halaman utama aplikasi untuk admin.



Gambar 3.30 Rancangan halaman utama aplikasi untuk admin.

### Halaman Kelola Data Sampah

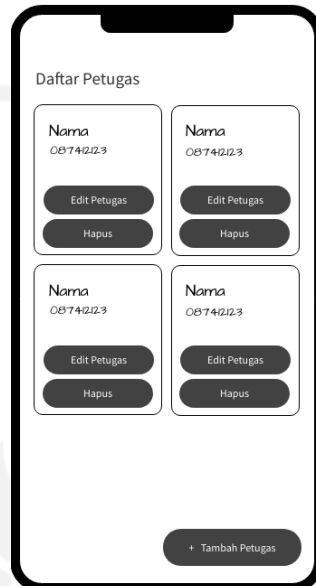
Pada halaman kelola data sampah menampilkan daftar kategori sampah. Admin bisa melakukan pengelolaan data sampah seperti tambah, hapus dan *update*. Gambar 3.31 menunjukkan rancangan antarmuka halaman kelola data sampah.



Gambar 3.31 Rancangan halaman kelola data sampah.

### Halaman Kelola Data Petugas

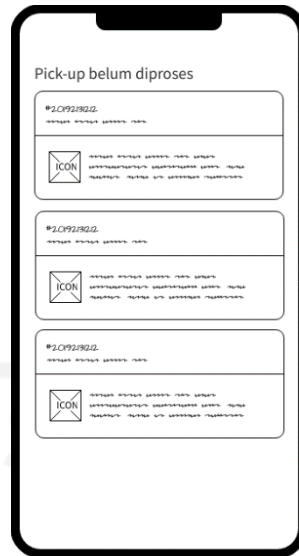
Pada halaman kelola data petugas menampilkan daftar petugas yang sudah terdaftar. Admin bisa melakukan pengelolaan data petugas seperti tambah, hapus dan *update*. Gambar 3.32 menunjukkan rancangan antarmuka halaman kelola data petugas.



Gambar 3.32 Rancangan halaman kelola data petugas.

### Halaman Transaksi Belum Diproses

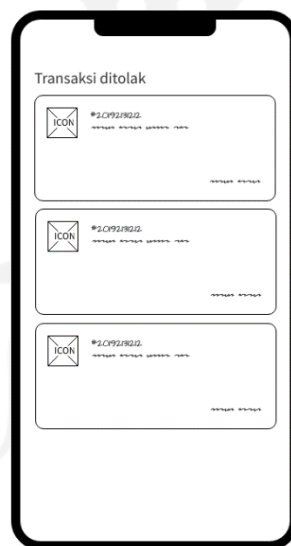
Pada halaman transaksi belum diproses menampilkan daftar permintaan sampah yang belum diproses oleh admin. Gambar 3.33 menunjukkan rancangan antarmuka halaman transaksi belum diproses.



Gambar 3.33 Rancangan halaman transaksi belum diproses.

### Halaman Permintaan Jemput Sampah Ditolak

Pada halaman permintaan jemput sampah ditolak menampilkan daftar permintaan sampah dengan status ditolak. Gambar 3.34 menunjukkan rancangan antarmuka halaman permintaan jemput sampah ditolak.

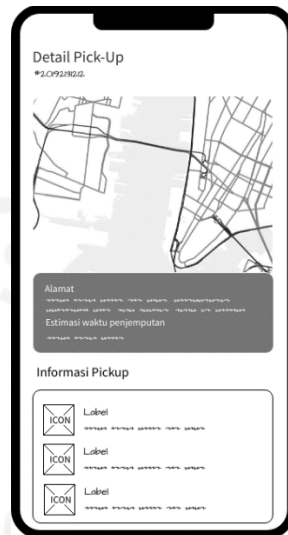


Gambar 3.34 Rancangan halaman permintaan jemput sampah ditolak.

### Halaman Detail *Pick-up*

Pada halaman detail *pick-up* menampilkan detail permintaan jemput sampah oleh masyarakat berupa data jarak antara lokasi pengelola sampah dengan lokasi titik jemput

sampah dan estimasi waktu penjemputan. Gambar 3.35 menunjukkan rancangan antarmuka halaman detail *pick-up*.



Gambar 3.35 Rancangan halaman detail *pick-up*.

### Halaman Utama Aplikasi untuk Petugas

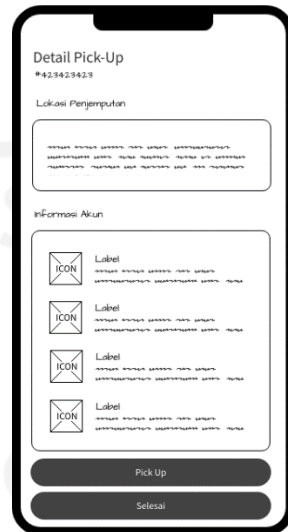
Pada halaman utama aplikasi untuk petugas menampilkan daftar permintaan jemput sampah yang sudah diproses oleh admin dan siap untuk dijemput oleh petugas. Gambar 3.36 menunjukkan rancangan antarmuka halaman utama aplikasi untuk petugas.



Gambar 3.36 Rancangan halaman utama aplikasi untuk petugas.

### Halaman Detail Permintaan Penjemputan

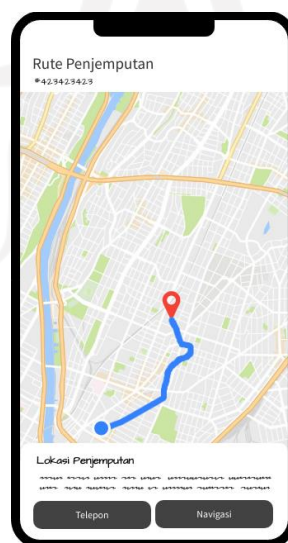
Pada halaman detail permintaan penjemputan menampilkan data alamat penjemputan dan informasi akun masyarakat. Gambar 3.37 menunjukkan rancangan halaman detail permintaan penjemputan.



Gambar 3.37 Rancangan halaman detail permintaan penjemputan.

### Halaman Rute Lokasi Penjemputan

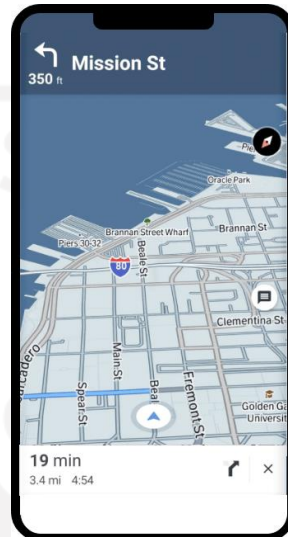
Pada halaman rute lokasi penjemputan menampilkan peta dengan rute yang telah ditentukan oleh sistem dari lokasi petugas berada menuju titik lokasi penjemputan. Gambar 3.38 menunjukkan rancangan antarmuka halaman rute lokasi penjemputan.



Gambar 3.38 Rancangan halaman rute lokasi penjemputan.

## Halaman Navigasi Menuju Lokasi Titik Penjemputan

Pada halaman navigasi menuju lokasi titik penjemputan akan ditampilkan menu navigasi menuju lokasi penjemputan. Di halaman ini juga menampilkan data jarak tempuh dan estimasi waktu menuju lokasi penjemputan. Gambar 3.39 menunjukkan rancangan antarmuka halaman navigasi menuju lokasi titik penjemputan.



Gambar 3.39 Rancangan halaman navigasi menuju lokasi titik penjemputan.

### 3.3 Implementasi

Tahapan implementasi dilakukan untuk mengimplementasikan semua hasil perancangan yang telah dilakukan untuk dikembangkan menjadi sebuah aplikasi. Pengembangan dilakukan dengan menggunakan Android Studio yang merupakan *environment* pengembangan terintegrasi besutan Google yang digunakan untuk mengembangkan aplikasi Android. Bahasa pemrograman yang akan digunakan yaitu menggunakan bahasa pemrograman Java. Untuk batas minimal perangkat yang didukung pada aplikasi yang akan dikembangkan yaitu perangkat Android versi 5.0 dengan kode nama *Lollipop*.

Pada aplikasi yang akan dikembangkan membutuhkan layanan untuk membaca atau menentukan titik lokasi penjemputan dari pengguna yaitu masyarakat. Untuk menentukan lokasi penjemputan dibutuhkan layanan dari Google bernama *Fused Location Provider API*. Untuk menggunakan *Fused Location Provider API* pada aplikasi perlu untuk mendefinisikan *permissions* pada *AndroidManifest.xml*, yaitu dengan menambahkan *ACCESS\_FINE\_LOCATION* untuk mengizinkan aplikasi mengakses lokasi melalui perangkat *smartphone*. Gambar 3.40 menunjukkan kode program untuk menambahkan *permissions*.

```

<manifest ... >
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
</manifest>

```

Gambar 3.40 Kode program menambahkan permissions.

AndroidManifest.xml sendiri adalah file yang harus ada disetiap proyek pengembangan android. AndroidManifest.xml berisi informasi penting mengenai aplikasi seperti nama paket aplikasi, komponen aplikasi dan izin yang diperlukan untuk mengakses sistem. Kemudian untuk mengetahui lokasi pengguna saat ini perlu membuat permintaan lokasi dengan memanfaatkan *LocationRequest*, *LocationRequest* merupakan objek yang digunakan untuk meminta kualitas layanan pembaruan lokasi dari *Fused Location Provider API*. Gambar 3.41 menunjukkan kode program untuk mengetahui lokasi pengguna saat ini.

```

LocationRequest locationRequest = new LocationRequest();
locationRequest.setInterval(10000);
locationRequest.setFastestInterval(3000);
locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
..
LocationServices.getFusedLocationProviderClient(WaitingListActivity.this)
    .requestLocationUpdates(locationRequest, new LocationCallback() {
@Override
public void onLocationResult(LocationResult locationResult) {
    super.onLocationResult(locationResult);
    LocationServices.getFusedLocationProviderClient(WaitingListActivity.this)
        .removeLocationUpdates(this);

    if (locationResult != null && locationResult.getLocations().size() > 0) {
        int latestLocationIndex = locationResult.getLocations().size() - 1;
        double latitude =
            locationResult.getLocations().get(latestLocationIndex).getLatitude();
        double longitude =
            locationResult.getLocations().get(latestLocationIndex).getLongitude();
    }
}
} Looper.getMainLooper());

```

Gambar 3.41 Kode program mengetahui lokasi pengguna.

Gambar 3.41 menunjukkan kode program untuk mendapatkan informasi lokasi pengguna berdasarkan titik koordinat *latitude* dan *longitude*. *Latitude* dan *longitude* didapatkan dari *method* “.getLatitude()” dan “.getLongitude()”. Pengguna juga bisa menentukan titik koordinat berdasarkan alamat yang telah diinputkan pengguna. Untuk bisa membaca titik koordinat membutuhkan *class* dari Java bernama *Geocoder*. Dengan *Geocoder* alamat yang telah



diinputkan pengguna bisa ditransformasikan menjadi titik koordinat *latitude* dan *longitude*. Gambar 3.42 menunjukkan kode program untuk mentransformasikan data alamat ke titik koordinat.

```
Geocoder geocoder = new Geocoder(WaitingListActivity.this, Locale.getDefault());

try {
    List addressList =
        geocoder.getFromLocationName(Common.currenUser.getHomeAddress(),1);
    if (addressList != null && addressList.size() > 0){
        Address address = (Address) addressList.get(0);
        double latNew = address.getLatitude();
        double langNew = address.getLongitude();
        lang.setText(String.valueOf(langNew));
        lat.setText(String.valueOf(latNew));
    }
} catch (IOException e) {
    e.printStackTrace();
}
```

Gambar 3.42 Kode program penggunaan geocoder.

Dari lokasi yang sudah diinput oleh masyarakat, admin dan petugas bisa mengetahui lokasi titik penjemputan sampah. Untuk membaca titik koordinat yang telah dikirimkan dibutuhkan layanan peta dan navigasi untuk menemukan titik lokasi dan rute perjalanan. Untuk memenuhi kebutuhan tersebut digunakan layanan dari MapBox. Layanan yang digunakan adalah *MapBox Maps* SDK dan *MapBox Navigation* SDK. *MapBox Maps* digunakan untuk melihat rute perjalanan yang akan ditempuh mulai dari posisi pengguna sampai lokasi tujuan. Sebelum mengimplementasikan *MapBox Maps* pada sistem diperlukan akses token, akses token bisa didapatkan ketika sudah memiliki akun MapBox. Kemudian token tersebut ditambahkan pada *strings.xml*. Gambar 3.43 merupakan kode program untuk menambahkan akses token.

```
...
<string name="mapbox_access_token">MAPBOX_ACCESS_TOKEN</string>
...
```

Gambar 3.43 Menambahkan akses token.

Setelah menambahkan akses token, kemudian perlu untuk menambahkan *library MapBox Maps* ke dependensi pada file *build.gradle (module-level)*. Hal ini dilakukan untuk bisa mengakses semua layanan yang ditawarkan oleh MapBox pada sistem yang akan dikembangkan. Gambar 3.44 menunjukkan *Library* yang ditambahkan pada dependensi.

```

...
dependencies {
    implementation 'com.mapbox.mapboxsdk:mapbox-android-sdk:9.6.1'
}
...

```

Gambar 3.44 Menambahkan library MapBox SDK.

Kemudian untuk menampilkan rute perjalanan dari lokasi pengguna saat ini menuju titik penjemputan perlu untuk *implements interface* bernama *OnMapReadyCallback* pada *activity* yang digunakan. *OnMapReadyCallback* berisi *method* “.onMapReady()” yang digunakan untuk mengatur bagaimana peta akan ditampilkan. *Method* tersebut akan dipanggil ketika peta sudah siap ditampilkan pada *activity* yang digunakan. Gambar 3.45 menunjukkan kode program untuk menampilkan peta pada sistem yang dikembangkan.

```

@Override
public void onMapReady(@NonNull MapboxMap mapboxMap) {
    this.mapboxMap = mapboxMap;
    mapboxMap.setStyle(Style.TRAFFIC_DAY, new Style.OnStyleLoaded() {
        @Override
        public void onStyleLoaded(@NonNull Style style) {
            enableLocationComponent(style);
        }
    });
}

```

Gambar 3.45 Kode program menampilkan peta *MapBox Maps*.

Gambar 3.45 menunjukkan bahwa pada *method* “.onMapReady()” terdapat *method-method* tambahan yang bisa ditambahkan secara opsional sesuai dengan apa yang dibutuhkan. *Method* “.setStyle()” digunakan untuk mengatur tampilan peta dan mengatur komponen apa saja yang akan ditampilkan pada peta ketika peta berhasil dimuat. Berikut merupakan beberapa kode program untuk menambahkan komponen yang akan ditampilkan pada peta.

```

...
@SuppressWarnings({"MissingPermission"})
private void enableLocationComponent(@NonNull Style loadedMapStyle) {
    locationComponent = mapboxMap.getLocationComponent();
    locationComponent.activateLocationComponent(this, loadedMapStyle);
    locationComponent.setLocationComponentEnabled(true);

    // Set the component's camera mode
    locationComponent.setCameraMode(CameraMode.TRACKING);
    locationComponent.setRenderMode(RenderMode.NORMAL);

    userLatitude = getIntent().getStringExtra("user_latitude");
    userLongitude = getIntent().getStringExtra("user_longitude");

    double newLatitude = Double.parseDouble(userLatitude);
    double newLongitude = Double.parseDouble(userLongitude);

    Point destinationPoint = Point.fromLngLat(newLongitude, newLatitude);

    Point originPoint =
    Point.fromLngLat(locationComponent.getLastKnownLocation()
    .getLongitude(),
    locationComponent.getLastKnownLocation().getLatitude());

    addDestinationIconSymbolLayer(loadedMapStyle);

    GeoJsonSource source =
        mapboxMap.getStyle().getSourceAs("destination-source-id");
    if (source != null) {
        source.setGeoJson(Feature.fromGeometry(destinationPoint));
    }

    getRoute(originPoint, destinationPoint);
...

```

Gambar 3.46 Kode program mengatur komponen pada peta.

Gambar 3.46 berisi kode program yang digunakan untuk mengatur bagaimana peta ditampilkan dan komponen apa saja yang akan ditampilkan. *Method* ".setCameraMode()" dan ".setRenderMode" digunakan untuk mengatur mode kamera pada peta yang akan ditampilkan. Mode kamera pada peta berguna untuk mengatur kamera pada saat peta dimuat dan mengikuti posisi pengguna saat ini. Kemudian ada *method* ".addDestinationIconSymbolLayer()" yang digunakan untuk menambahkan komponen *icon* penanda pada peta. Yang terakhir ada *method* ".getRoute()" yang berisi 2 *argument* yaitu *originPoint* dan *destinationPoint*. *Argument* *originPoint* merupakan objek dari *class* bernama *Point* bawaan dari *MapBox* yang berisi lokasi pengguna saat ini yang direpresentasikan dengan titik koordinat *latitude* dan *longitude*. Kemudian *destinationPoin* berisi lokasi titik koordinat penjemputan atau titik koordinat tujuan. Gambar 3.47 menunjukkan isi dari *method* ".getRoute()" .

```

private void getRoute(Point origin, Point destination) {
    NavigationRoute.builder(this)
        .accessToken(Mapbox.getAccessToken())
        .origin(origin)
        .destination(destination)
        .build()
        .getRoute(new Callback<DirectionsResponse>() {
            @Override
            public void onResponse(Call<DirectionsResponse> call,
                Response<DirectionsResponse> response) {
                if (response.body() == null) {
                    Log.e(TAG, "No routes found,
                        make sure you set the right user and access token.");
                    return;
                } else if (response.body().routes().size() < 1) {
                    Log.e(TAG, "No routes found");
                    return;
                }

                currentRoute = response.body().routes().get(0);

                // Draw the route on the map
                if (navigationMapRoute != null) {
                    navigationMapRoute.removeRoute();
                } else {
                    navigationMapRoute = new NavigationMapRoute(null, mapView,
                        mapboxMap, R.style.NavigationMapRoute);
                }

                navigationMapRoute.addRoute(currentRoute);

                @Override
                public void onFailure(Call<DirectionsResponse> call,
                    Throwable throwable) {
                    Log.e(TAG, "Error: " + throwable.getMessage());
                }
            });
}

```

Gambar 3.47 Kode program *method* `getRoute()`.

Untuk menampilkan navigasi dengan *MapBox Navigation SDK* cukup panggil *class* `NavigationLauncher` pada *method* “`.onStyleLoaded()`” di dalam *method* “`.onMapReady()`”. Gambar 3.48 menunjukkan kode program untuk menampilkan navigasi pada aplikasi.

```

...
@Override
public void onMapReady(@NonNull MapboxMap mapboxMap) {
    this.mapboxMap = mapboxMap;
    mapboxMap.setStyle(Style.TRAFFIC_DAY, new Style.OnStyleLoaded() {
        @Override
        public void onStyleLoaded(@NonNull Style style) {
            enableLocationComponent(style);
            btnNavigate.setOnClickListener(new View.OnClickListener() {
                @Override public void onClick(View v) {
                    boolean simulateRoute = false;
                    NavigationLauncherOptions options =
                        NavigationLauncherOptions.builder()
                            .directionsRoute(currentRoute)
                            .shouldSimulateRoute(simulateRoute)
                            .build();

                    NavigationLauncher.startNavigation(MapViewActivity.this,
                        options);
                }
            });
        }
    });
}
...

```

Gambar 3.48 Kode program menampilkan navigasi *MapBox SDK*.

Dengan ditabakkannya layanan *MapBox Maps SDK* dan *MapBox Navigation SDK* pada sistem yang dikembangkan, kini admin dan petugas bisa mengetahui titik lokasi penjemputan sampah yang telah dikirim oleh masyarakat.

### 3.4 Pengujian

Pengujian yang dilakukan pada penelitian ini adalah dengan menggunakan metode *Black Box testing* untuk menguji fungsionalitas yang ada pada aplikasi dan pengujian usability untuk mengukur tingkat kenyamanan pengguna dalam menggunakan aplikasi yang telah dikembangkan.

#### 3.4.1 Pengujian Black Box

Pengujian *black box* dilakukan untuk menguji perangkat lunak dari segi fungsionalitas pada aplikasi yang dikembangkan dengan skenario-skenario yang diberikan. Pengujian *black box* dilakukan berdasarkan pandangan dari pengguna akhir sebagai pengguna aplikasi. *Black box* perlu dilakukan untuk menangani masukan dari pengguna baik masukan yang valid atau tidak valid. Untuk pengujian akan dilakukan pada ke 3 aplikasi yang telah dikembangkan yaitu pada aplikasi untuk masyarakat, aplikasi untuk admin dan aplikasi untuk petugas.

### 3.4.2 Pengujian Usabilitas

Pengujian usabilitas dilakukan untuk mengukur seberapa besar tingkat kenyamanan pengguna dalam menggunakan aplikasi. Terdapat beberapa indikator yang akan digunakan guna untuk melakukan penilaian pada pengujian usabilitas ini. Tabel 3.1 menunjukkan indikator pada pengujian yang akan dilakukan.

Tabel 3.1 Indikator penilaian pengujian usabilitas

<b>Indikator</b>	<b>Penjelasan</b>
Efe	Efektifitas
Efi	Efisiensi
Lea	Mudah dipahami pengguna
MeR	Minim kesalahan
Sat	Kepuasan dan kenyamanan

Dari indikator yang sudah ada dapat dibuat beberapa pertanyaan yang sesuai dengan indikator yang ada. Pertanyaan akan dibuat menjadi sebuah formulir kuesioner yang akan digunakan sebagai media untuk melakukan pengujian kepada *end-user*. Pengujian akan dilakukan pada aplikasi untuk masyarakat dengan skenario pengujian mulai dari melakukan pendaftaran pengguna baru sampai dengan mengirim permintaan penjemputan sampah. Setelah selesai melakukan pengujian terhadap aplikasi yang telah dikembangkan, masyarakat akan diminta untuk mengisi kuesioner yang akan diberikan untuk menilai seberapa nyaman tingkat kenyamanan pengguna dalam menggunakan aplikasi. Tabel 3.2 merupakan rancangan daftar pertanyaan yang akan digunakan untuk melakukan pengujian usabilitas pada aplikasi yang telah dikembangkan.

Tabel 3.2 Rancangan daftar pertanyaan kuesioner

No	Indikator	Pertanyaan	TS	KS	CS	S	SS
1	Lea	Apakah aplikasi memiliki tampilan yang mudah dipahami dan mudah digunakan?					
2		Apakah aplikasi menggunakan bahasa percakapan yang mudah dipahami?					
3	Efe	Apakah aplikasi mampu memahami masukan pengguna dalam melakukan permintaan penjemputan sampah?					
4		Apakah aplikasi mampu memberikan respons yang sesuai dengan yang diharapkan?					
5	MeR	Apakah aplikasi tidak membingungkan saat digunakan?					
6		Apakah aplikasi minim memberikan kesalahan saat merespons permintaan pengguna?					
7	Efi	Apakah aplikasi dapat memberikan manfaat kepada pengguna dalam melakukan permintaan penjemputan sampah?					
8		Apakah aplikasi dapat memproses permintaan penjemputan sampah yang dilakukan oleh pengguna?					
9	Sat	Apakah dengan adanya aplikasi ini untuk melakukan penjemputan sampah dapat memberikan kemudahan kepada pengguna dalam pengelolaan sampah?					
10		Apakah pengguna puas dengan adanya aplikasi ini untuk melakukan permintaan jemput sampah?					

Berdasarkan dari rancangan daftar pertanyaan kuesioner, berikut penjelasan dari pilihan jawaban yang telah disediakan pada kuesioner tersebut:

- a. TS = Tidak Setuju (1 poin)
- b. KS = Kurang Setuju (2 poin)
- c. CS = Cukup Setuju (3 poin)
- d. S = Setuju (4 poin)
- e. SS = Sangat Setuju (5 poin)

Rumus perhitungan dalam pengujian ini terdiri dari 2 tahapan, yang pertama menentukan nilai pengujian yang dapat dilihat pada persamaan (3.1) dan menentukan nilai skor usabilitas yang dapat diketahui pada persamaan (3.2). Berikut rumus masing-masing persamaan:

a. Nilai pengujian

$$P_i = \frac{\sum_1^5 b_n \cdot h_n}{m} \quad (3.1)$$

$i$  = nomor pertanyaan

$b$  = bobot poin

$h$  = banyak isian

$m$  = jumlah responden

b. Skor pengujian usabilitas

$$P_i = \frac{\sum P_i}{\text{Jumlah pertanyaan}} \quad (3.2)$$

$\sum P_i$  = Jumlah nilai pengujian

Setelah melakukan perhitungan dengan menggunakan persamaan di atas, maka akan diketahui nilai pengujian dan skor pengujian. Nilai pengujian dan skor pengujian akan digunakan untuk mengetahui seberapa layak aplikasi yang telah dikembangkan. Tabel 3.3 menunjukkan keterangan dari nilai pengujian dan skor pengujian yang telah didapat dalam proses pengujian usabilitas.

Tabel 3.3 Keterangan nilai pengujian usabilitas

Nilai	Keterangan
1,0 – 1,5	Tidak Layak
1,6 – 2,5	Kurang Layak
2,6 – 3,5	Cukup Layak
3,6 – 4,5	Layak
4,6 – 5,0	Sangat layak



## BAB IV

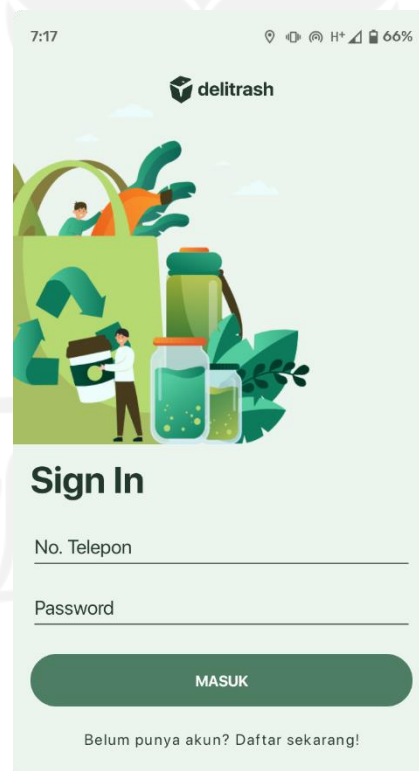
### HASIL DAN PEMBAHASAN

#### 4.1 Hasil Implementasi Sistem

Penelitian yang dilakukan telah berhasil mengimplementasikan semua tahapan dan telah berhasil mengembangkan aplikasi jemput sampah berbasis Android. Sesuai yang telah tertulis pada tahapan implementasi, penelitian ini mengembangkan 3 buah aplikasi yaitu aplikasi untuk masyarakat, aplikasi untuk admin dan aplikasi untuk petugas.

##### 4.1.1 Halaman Login

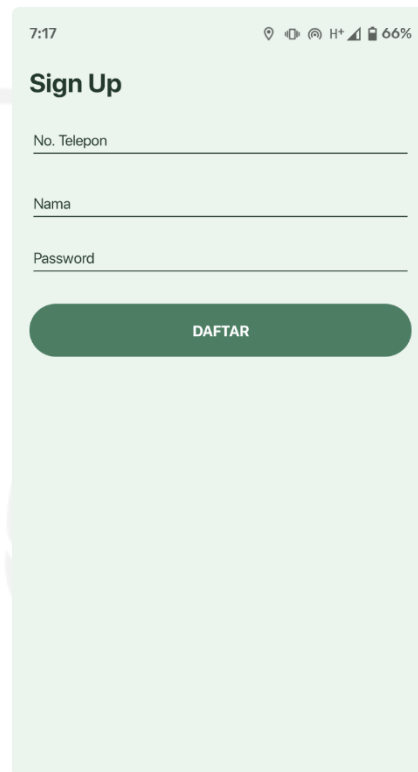
Gambar 4.1 menunjukkan tampilan dari halaman login pada aplikasi untuk masyarakat yang digunakan untuk mengisi data nomor telepon dan password agar bisa masuk ke dalam halaman menu utama dan mengakses fitur-fitur yang ada di dalam aplikasi.



Gambar 4.1 Halaman login.

#### 4.1.2 Halaman Daftar Akun

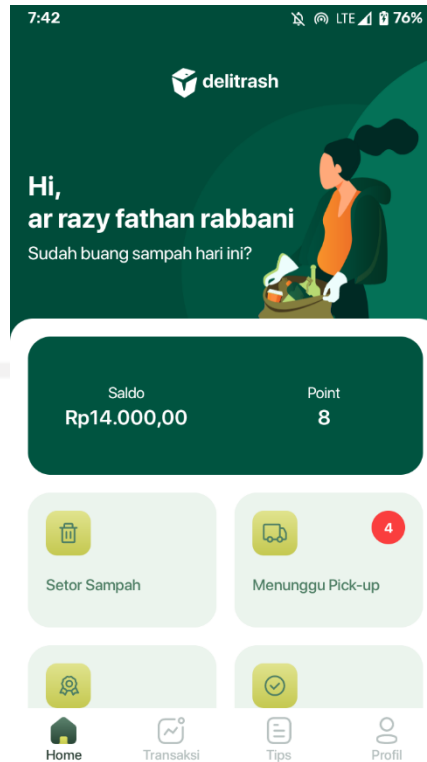
Gambar 4.2 menunjukan tampilan antarmuka halaman daftar akun yang digunakan untuk masyarakat yang belum memiliki akun. Masyarakat bisa mendaftar akun dengan memasukan data berupa nomor telepon, nama dan password.



Gambar 4.2 Halaman daftar akun.

#### 4.1.3 Halaman Utama Aplikasi untuk Masyarakat

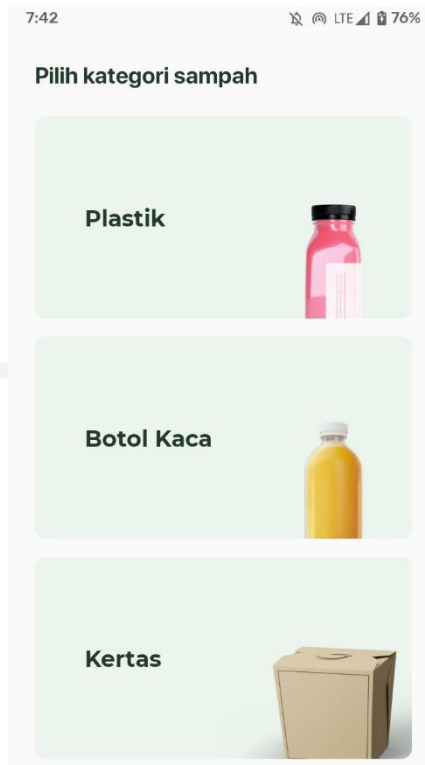
Gambar 4.3 menunjukkan tampilan antarmuka pada halaman utama aplikasi untuk masyarakat. Pada menu utama menyajikan beberapa informasi mengenai pengguna diantaranya nama pengguna, saldo yang dimiliki dan juga point yang sudah dikumpulkan. Terdapat juga beberapa menu navigasi untuk menuju halaman lain untuk memudahkan masyarakat dalam menggunakan aplikasi.



Gambar 4.3 Halaman menu utama pada aplikasi masyarakat.

#### 4.1.4 Halaman Kategori Sampah

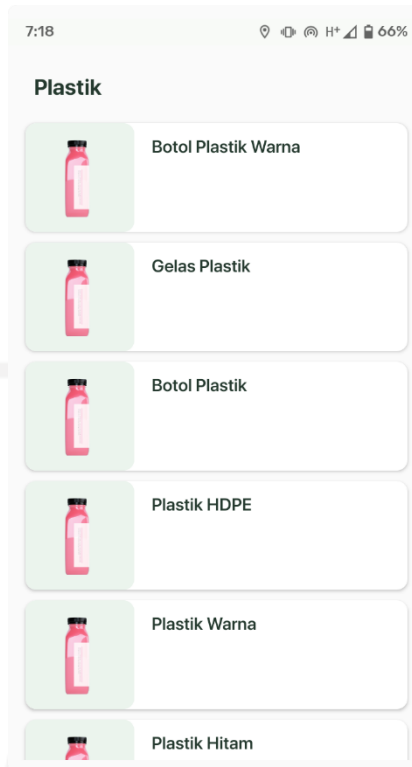
Gambar 4.4 menunjukkan tampilan antarmuka halaman kategori sampah. Terdapat beberapa kategori sampah yang ada pada aplikasi diantaranya sampah plastik, botol kaca, kertas, logam dan sampah elektronik. Halaman kategori sampah digunakan untuk mengkategorikan sampah supaya masyarakat lebih mudah untuk mencari jenis sampah yang ingin ditambahkan ke antrian.



Gambar 4.4 Halaman kategori sampah.

#### 4.1.5 Halaman Daftar Sampah

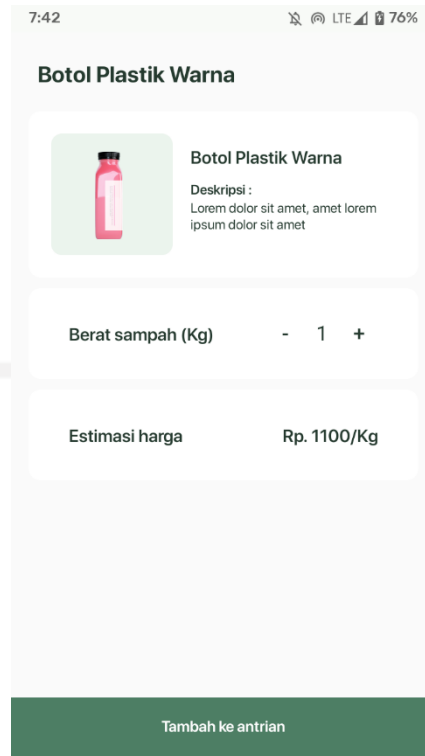
Gambar 4.5 menunjukkan tampilan antarmuka pada halaman daftar sampah. Pada halaman daftar sampah menampilkan daftar data sampah berupa gambar sampah dan nama sampah. Daftar sampah ditampilkan berdasarkan kategori sampah yang dipilih pengguna pada halaman sebelumnya.



Gambar 4.5 Halaman daftar sampah.

#### 4.1.6 Halaman Detail Sampah

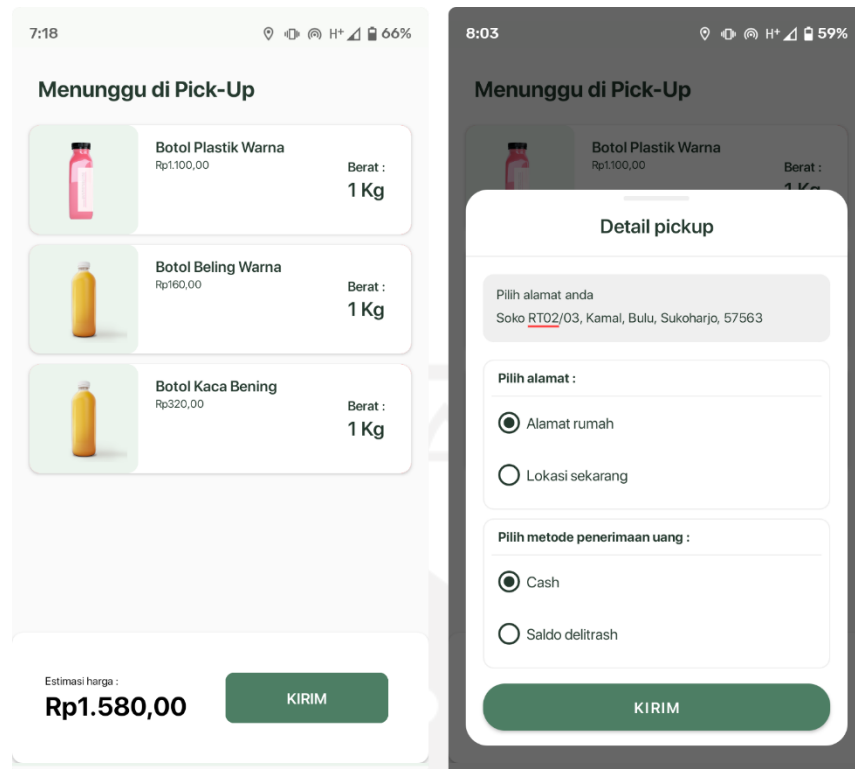
Gambar 4.6 menunjukkan tampilan antarmuka pada halaman detail sampah. Halaman detail sampah menampilkan data informasi detail sampah yang dipilih oleh masyarakat. Informasi detail sampah meliputi nama sampah, gambar sampah dan deskripsi sampah. Terdapat juga informasi tambahan yaitu berat sampah dan estimasi harga sampah per kilogram. Berat sampah digunakan untuk menentukan berat sampah yang ingin dikirimkan oleh masyarakat.



Gambar 4.6 Halaman detail sampah.

#### 4.1.7 Halaman Menunggu *Pick-up*

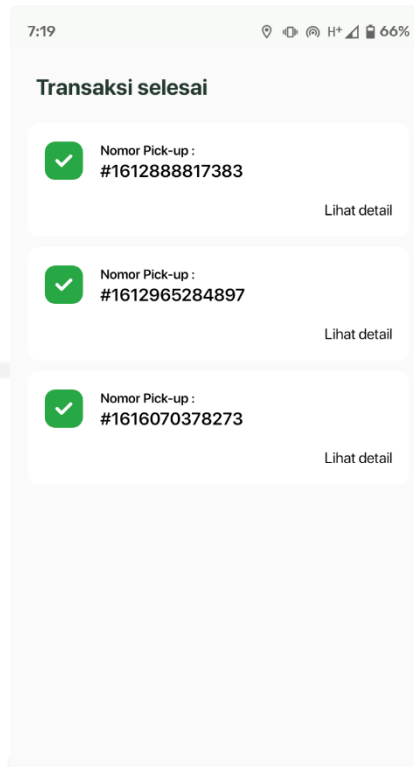
Gambar 4.7 menunjukkan tampilan antarmuka pada halaman menunggu *pick-up*. Halaman menunggu *pick-up* berisi data antrean sampah yang sudah ditambahkan oleh masyarakat. Sampah yang ada pada antrean akan dikalkulasika estimasi total harganya berdasarkan jumlah berat sampah. Kemudian pada halaman menunggu *pick-up* terdapat tombol kirim yang digunakan untuk memasukan informasi tambahan berupa alamat dan juga penerimaan uang. Namun penerimaan uang dengan opsi saldo belum diimplementasikan pada sistem, opsi penerimaan uang yang tersedia hanya berupa uang tunai. Setelah masyarakat selesai memasukan data tambahan dan menekan tombol kirim maka sistem akan mengirimkan notifikasi kepada admin.



Gambar 4.7 Halaman menunggu *pick-up*.

#### 4.1.8 Halaman Transaksi Selesai

Gambar 4.8 menunjukkan tampilan antarmuka pada halaman transaksi selesai. Halaman transaksi selesai menampilkan daftar transaksi yang sudah selesai atau permintaan jemput sampah sudah terpenuhi.

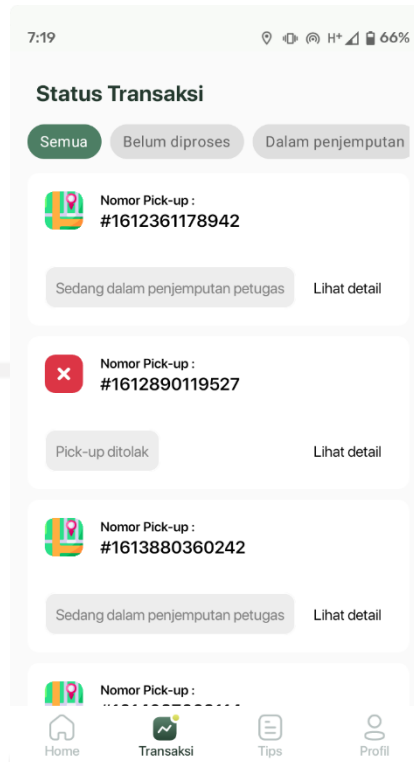


Gambar 4.8 Halaman transaksi selesai.

#### 4.1.9 Halaman Transaksi

Gambar 4.9 menunjukkan tampilan antarmuka halaman transaksi. Halaman transaksi menampilkan daftar transaksi atau permintaan jemput sampah yang telah dibuat oleh masyarakat, daftar transaksi yang ditampilkan disajikan berdasarkan status transaksinya. Status transaksi dibagi menjadi 3 yaitu transaksi belum diproses, transaksi dalam penjemputan petugas dan transaksi ditolak. Pengguna bisa menampilkan semua transaksi yang telah dilakukan.





Gambar 4.9 Halaman transaksi.

#### 4.1.10 Halaman Detail Transaksi

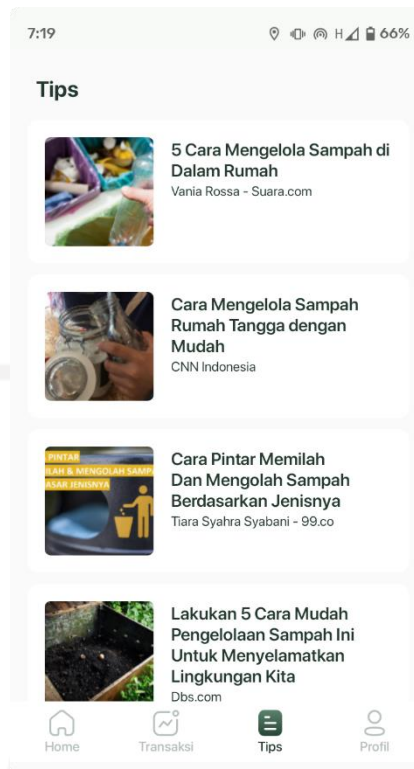
Gambar 4.10 menunjukkan tampilan antarmuka halaman detail transaksi permintaan jemput sampah. Pada halaman detail transaksi menampilkan beberapa informasi terkait permintaan jemput sampah seperti nomor id transaksi, status penjemputan, nama petugas, informasi akun dan informasi penjemputan. Informasi penjemputan berisi daftar sampah yang dikirimkan dan sudah dikalkulasikan total estimasi harganya.



Gambar 4.10 Halaman detail transaksi.

#### 4.1.11 Halaman Tips

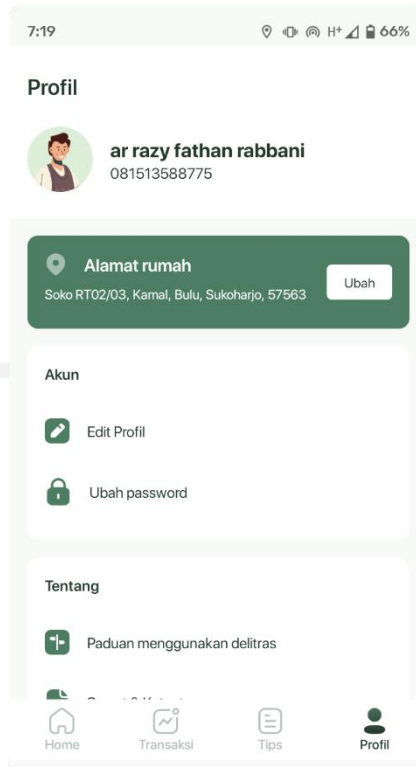
Gambar 4.11 menunjukkan tampilan antarmuka halaman tips. Halaman tips digunakan untuk menampilkan daftar artikel pilihan tentang tips pengelolaan sampah.



Gambar 4.11 Halaman tips.

#### 4.1.12 Halaman Profil

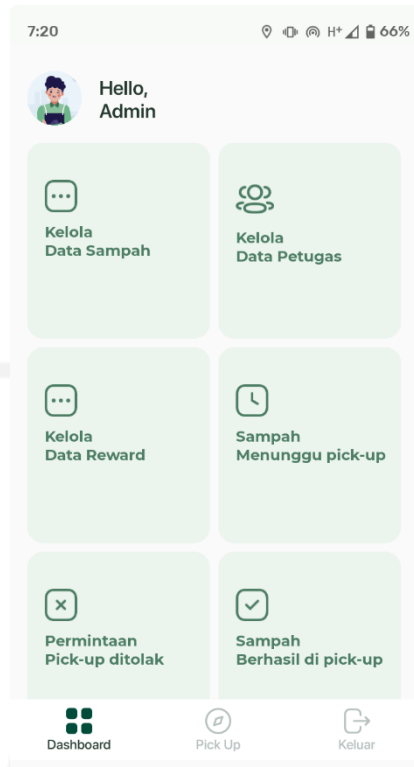
Gambar 4.12 menunjukkan tampilan antarmuka halaman profil. Halaman profil berisi informasi mengenai akun seperti nama pengguna, nomor telepon pengguna dan juga alamat pengguna. Pada halaman profil pengguna juga bisa melakukan penggantian data alamat, *username* dan *password*.



Gambar 4.12 Halaman profil.

#### 4.1.13 Halaman Utama Aplikasi untuk Admin

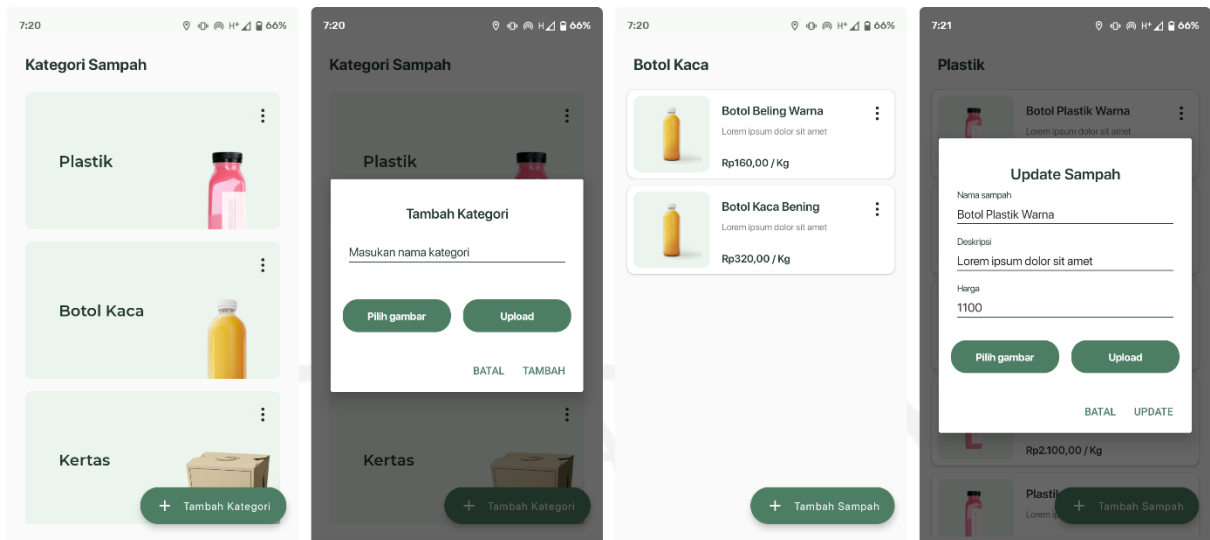
Gambar 4.13 menunjukkan tampilan antarmuka halaman utama aplikasi untuk admin. Halaman utama aplikasi untuk admin ini menampilkan menu-menu untuk memudahkan admin berpindah dari menu satu ke menu yang lainnya. Menu yang ada pada halaman utama pada aplikasi untuk admin diantaranya menu kelola data sampah, menu kelola data petugas, menu sampah menunggu *pick-up*, menu permintaan *pick-up* ditolak dan menu sampah berhasil di *pick-up*.



Gambar 4.13 Halaman utama aplikasi untuk admin.

#### 4.1.14 Halaman Kelola Data Sampah

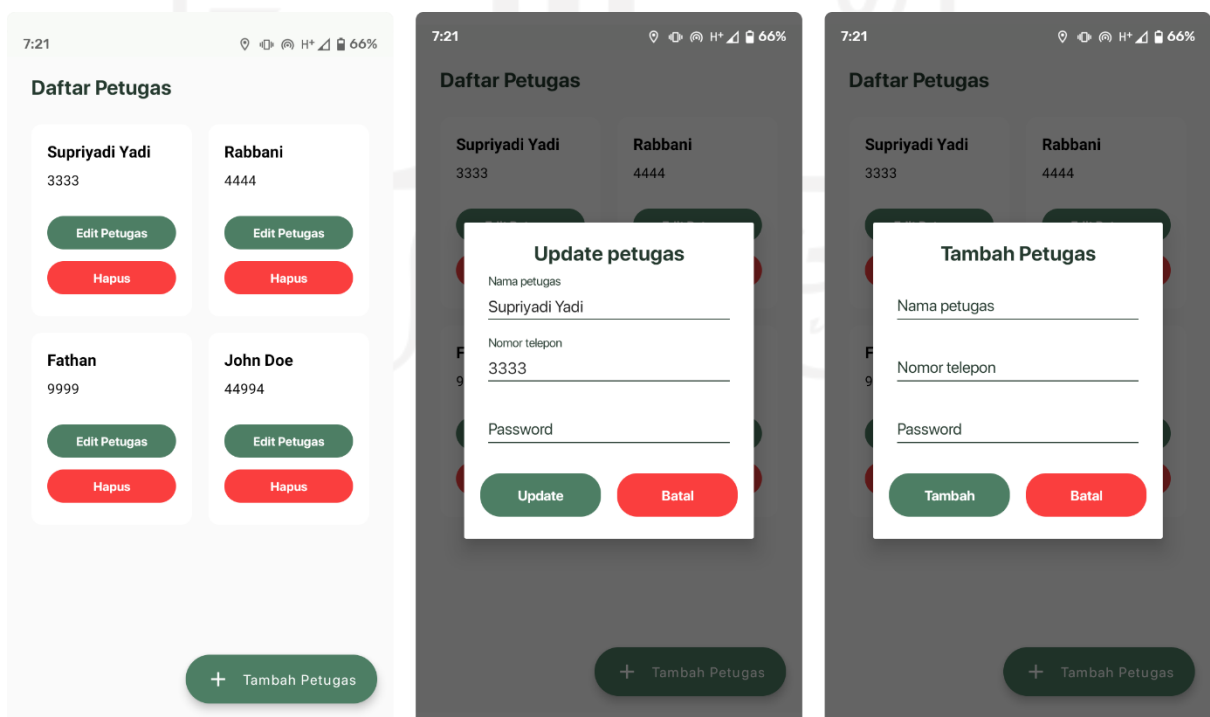
Gambar 4.14 menunjukkan tampilan antarmuka halaman kelola data sampah. Pada halaman kelola data sampah admin bisa melakukan pengelolaan data sampah seperti menambahkan kategori sampah, mengubah data kategori sampah dan menghapus kategori sampah. Admin juga bisa menambahkan sampah pada kategori sampah, menghapus dan merubah data sampah. Data sampah yang bisa diubah adalah gambar sampah, nama sampah, deskripsi sampah dan harga sampah.



Gambar 4.14 Halaman kelola data sampah.

#### 4.1.15 Halaman Kelola Data Petugas

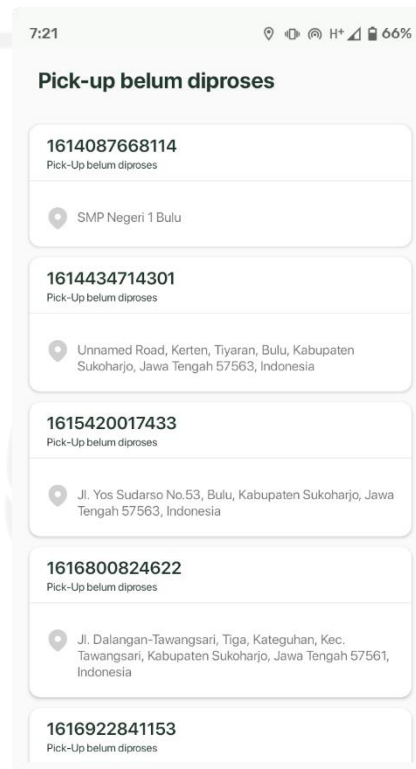
Gambar 4.15 menunjukkan tampilan antarmuka halaman kelola data petugas. Halaman kelola data petugas digunakan untuk admin mengelola data petugas. Admin bisa melakukan pengelolaan data petugas seperti menambahkan petugas baru, mengubah data petugas dan juga menghapus data petugas. Data yang dapat diubah oleh admin adalah nama petugas, nomor petugas dan password. Data nomor dan password nantinya akan digunakan oleh petugas untuk masuk ke dalam aplikasi untuk petugas.



Gambar 4.15 Halaman kelola data petugas.

#### 4.1.16 Halaman Transaksi Belum Diproses

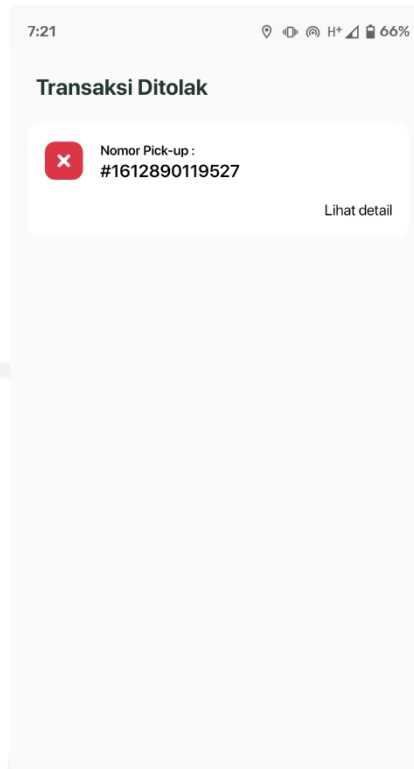
Gambar 4.16 menunjukkan tampilan antarmuka halaman transaksi belum diproses. Halaman transaksi belum diproses menampilkan daftar transaksi permintaan jemput sampah yang sudah dikirimkan oleh masyarakat dan belum diproses oleh admin. Diproses disini maksudnya adalah apakah permintaan akan ditolak atau diterima dan diteruskan ke petugas untuk bisa dijemput.



Gambar 4.16 Halaman kelola data petugas.

#### 4.1.17 Halaman Permintaan Jemput Sampah Ditolak

Gambar 4.17 menunjukkan tampilan antarmuka halaman permintaan jemput sampah ditolak. Halaman permintaan jemput sampah ditolak berisi daftar permintaan penjemputan sampah yang dilakukan oleh masyarakat dan ditolak oleh admin.

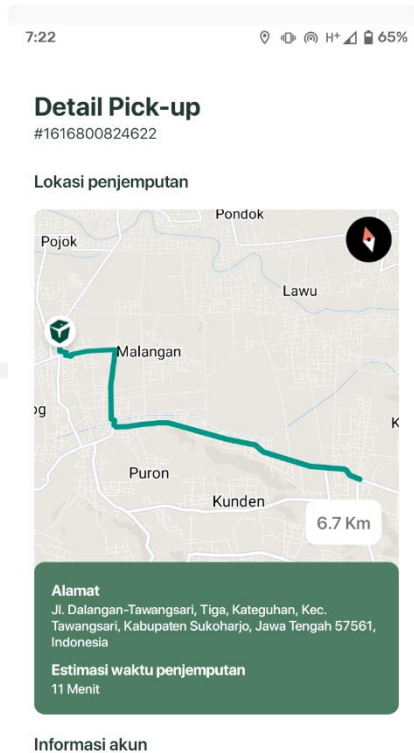


Gambar 4.17 Halaman permintaan jemput sampah ditolak.

#### 4.1.18 Halaman Detail *Pick-up*

Gambar 4.18 menunjukan tampilan antarmuka halaman detail *pick-up* pada aplikasi untuk admin. Pada halaman detail *pick-up*, admin bisa melihat informasi mengenai permintaan jemput sampah yang dilakukan oleh masyarakat. Informasi yang disajikan pada halaman detail *pick-up* berupa lokasi penjemputan, informasi akun dan informasi *pick-up*. Informasi penjemputan menampilkan peta dan rute penjemputan. Dari peta yang ditampilkan admin bisa melihat alamat lengkap lokasi penjemputan, jarak lokasi penjemputan dan juga estimasi waktu penjemputan.

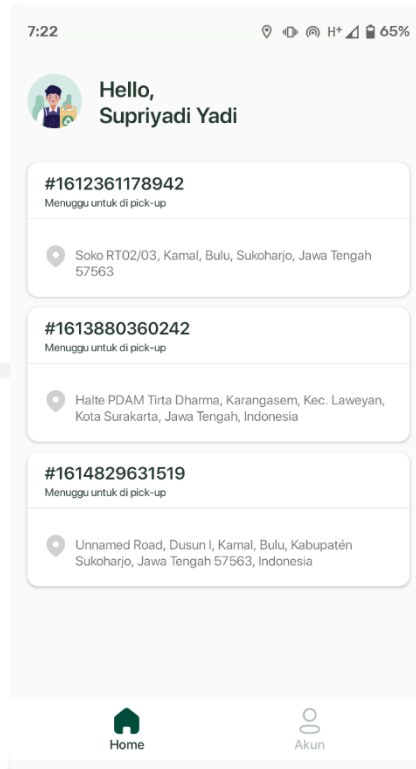




Gambar 4.18 Halaman detail *pick-up*.

#### 4.1.19 Halaman Utama Aplikasi untuk Petugas

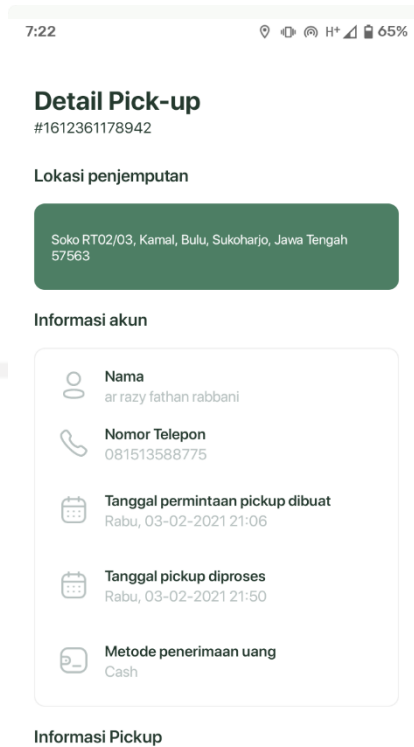
Gambar 4.19 menunjukkan tampilan antarmuka halaman utama aplikasi untuk petugas. Halaman utama aplikasi untuk petugas menampilkan daftar permintaan penjemputan sampah oleh masyarakat yang sudah diproses oleh admin.



Gambar 4.19 Halaman halaman utama aplikasi untuk petugas.

#### 4.1.20 Halaman Detail Permintaan Penjemputan

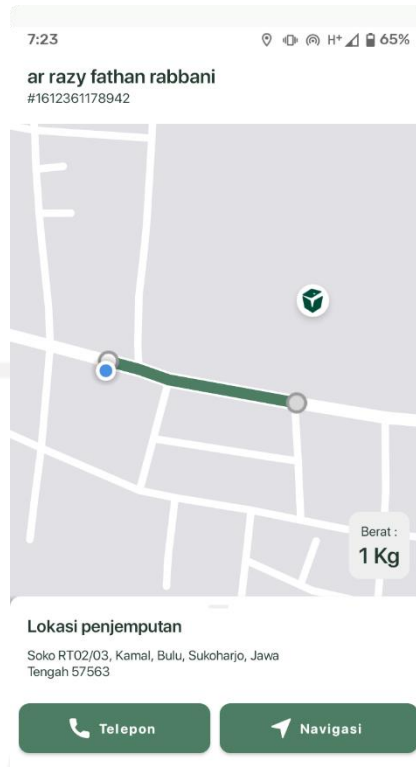
Gambar 4.20 menunjukkan tampilan antarmuka halaman detail permintaan penjemputan. Halaman detail permintaan penjemputan menampilkan informasi mengenai permintaan penjemputan sampah seperti alamat penjemputan, informasi akun dan informasi *pick-up*.



Gambar 4.20 Halaman detail permintaan penjemputan.

#### 4.1.21 Halaman Rute Lokasi Penjemputan

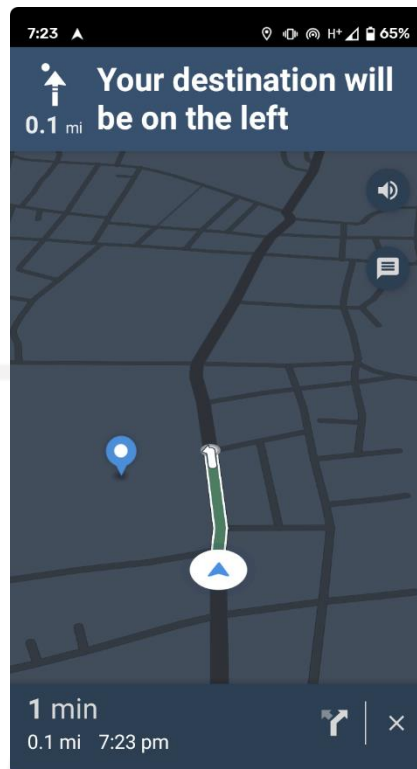
Gambar 4.21 menunjukkan tampilan antarmuka halaman rute lokasi penjemputan. Halaman rute penjemputan menampilkan peta rute antara lokasi petugas saat ini dan lokasi titik penjemputan sampah secara *realtime*. Pada peta juga ditampilkan penanda titik lokasi tempat penjemputan yang dikirimkan oleh masyarakat. Halaman rute lokasi penjemputan terdapat fitur telepon dan navigasi. Fitur telepon bisa digunakan oleh petugas jika kesulitan mencari lokasi penjemputan. Kemudian fitur navigasi bisa digunakan oleh petugas untuk mempermudah dalam perjalanan menuju ke lokasi titik penjemputan.



Gambar 4.21 Halaman rute lokasi penjemputan.

#### 4.1.22 Halaman Navigasi Menuju Lokasi Titik Penjemputan

Gambar 4.22 menunjukkan tampilan antarmuka navigasi menuju lokasi titik penjemputan. Halaman navigasi menuju lokasi titik penjemputan menampilkan navigasi yang dapat membantu petugas dalam perjalanan penjemputan sampah. Navigasi menampilkan informasi penunjuk arah *turn-by-turn* menuju lokasi titik penjemputan. Pada navigasi juga ditampilkan jarak yang ditempuh dan estimasi waktu perjalanan.



Gambar 4.22 Halaman navigasi menuju lokasi titik penjemputan.

## 4.2 Hasil Pengujian

### 4.2.1 Hasil Pengujian Black Box Aplikasi untuk masyarakat

Hasil pengujian fungsionalitas dengan metode *black box* pada aplikasi untuk masyarakat menunjukkan bahwa semua fungsi dan fitur yang ada di dalam aplikasi untuk masyarakat dari pertama kali dijalankan sampai selesai dapat berjalan dengan baik sesuai dengan apa yang telah direncanakan. Ada terdapat 14 fitur yang sudah diujikan, Tabel 4.1 menunjukkan hasil pengujian pada aplikasi untuk masyarakat.

Tabel 4.1 Hasil pengujian *black box* aplikasi untuk masyarakat.

No	Fitur	Fungsi yang diharapkan	Status
1	Login	Login dengan nomor telepon dan <i>password</i> yang sudah terdaftar.	Sukses
2	Daftar akun	Daftar akun pengguna baru dengan memasukan nomor telepon, nama dan <i>password</i> .	Sukses
3	Menu kategori sampah	Menampilkan nama dan gambar kategori sampah.	Sukses
4	Menu daftar sampah	Menampilkan data sampah berupa nama dan gambar berdasarkan kategori sampah.	Sukses
5	Tambah sampah ke antrean	Menambahkan data sampah ke antrean sebelum dikirim ke admin.	Sukses
6	Hapus sampah dari antrean	Data sampah yang dipilih dari antrean terhapus.	Sukses
7	Pilih lokasi penjemputan	Input data lokasi oleh masyarakat. Bisa memilih lokasi sekarang atau alamat rumah.	Sukses
8	Kirim permintaan penjemputan sampah	Muncul <i>dialog</i> permintaan penjemputan sampah telah terkirim.	Sukses
9	Menu transaksi	Menampilkan semua daftar permintaan penjemputan sampah.	Sukses
10	Detail transaksi	Menampilkan informasi detail penjemputan sampah.	Sukses
11	Menu transaksi berhasil	Menampilkan daftar transaksi penjemputan sampah yang berhasil dilakukan oleh petugas.	Sukses
12	Menu tips	Menampilkan daftar artikel terkait tips pengelolaan sampah.	Sukses
13	Menu profil	Menampilkan informasi pengguna berupa nama, nomor telepon dan alamat.	Sukses
14	Ubah profil	Data nama, alamat dan <i>password</i> berubah dengan nama, alamat dan <i>password</i> yang baru.	Sukses

#### 4.2.2 Hasil Pengujian Black Box Aplikasi untuk Admin

Hasil pengujian *black box* pada aplikasi untuk admin menunjukkan bahwa semua fungsionalitas yang ada pada aplikasi berjalan dengan baik dan sesuai dengan apa yang diharapkan. Ada terdapat 8 fitur yang telah diujikan pada pengujian ini, Tabel 4.2 menunjukkan hasil pengujian pada aplikasi untuk admin.

Tabel 4.2 Hasil pengujian *black box* aplikasi untuk admin.

No	Fitur	Fungsi yang diharapkan	Status
1	Login	Login dengan memasukkan nomor telepon dan password yang sudah terdaftar.	Sukses
2	Kelola data sampah oleh admin	Menambahkan, mengubah dan menghapus data kategori sampah dan sampah.	Sukses
3	Kelola data petugas oleh admin	Menambah, mengubah dan menghapus data petugas.	Sukses
4	Menu permintaan jemput sampah	Menampilkan semua permintaan penjemputan sampah dari pengguna.	Sukses
5	Detail permintaan penjemputan sampah	Menampilkan informasi detail penjemputan sampah berupa alamat, jarak lokasi, estimasi waktu penjemputan dan informasi pengguna.	Sukses
6	Ubah status permintaan penjemputan sampah	Status penjemputan berubah sesuai dengan input oleh admin.	Sukses
7	Menu sampah berhasil dijemput	Menampilkan semua daftar permintaan penjemputan sampah oleh pengguna yang berhasil.	Sukses
8	Menu permintaan penjemputan sampah ditolak	Menampilkan semua daftar permintaan penjemputan sampah oleh pengguna yang ditolak.	Sukses

#### 4.2.3 Hasil Pengujian Black Box Aplikasi untuk Petugas

Hasil pengujian *black box* pada aplikasi untuk petugas menunjukkan bahawa semua fungsionalitas yang ada pada aplikasi berjalan dengan baik dan sesuai dengan apa yang diharapkan. Ada terdapat 6 fitur yang telah diujikan pada pengujian ini, Tabel 4.3 menunjukkan hasil pengujian pada aplikasi untuk petugas.

Tabel 4.3 Hasil pengujian *black box* aplikasi untuk petugas.

No	Fitur	Fungsi yang diharapkan	Status
1	Login	Login dengan memasukkan nomor telepon dan password yang sudah terdaftar.	Sukses
2	Menu halaman utama petugas	Menampilkan semua daftar permintaan penjemputan sampah yang sudah diproses admin.	Sukses
3	Menu lihat rute	Menampilkan peta dan rute lokasi petugas menuju lokasi titik penjemputan sampah.	Sukses
5	Telepon masyarakat oleh petugas	Melakukan panggilan telepon melalui aplikasi ke pengguna.	Sukses
6	Menu navigasi	Menampilkan navigasi dari lokasi petugas menuju titik penjemputan sampah	Sukses

#### 4.2.4 Hasil Pengujian Usabilitas

Pengujian usabilitas dilakukan kepada 15 pengguna yang merupakan pengguna sebagai masyarakat atau mahasiswa. Pengujian dilakukan dengan cara meminta pengguna mencoba menggunakan aplikasi yang dikembangkan dengan skenario yang sudah ditetapkan. Kemudian setelah mencoba aplikasi, kemudian pengguna mengisi kuesioner untuk menjawab beberapa pertanyaan yang akan digunakan untuk penilaian dalam pengujian usabilitas. Penilaian akan menentukan apakah aplikasi sudah memenuhi tingkat kenyamanan pengguna dalam menggunakan aplikasi atau belum. Tabel 4.4 menunjukkan hasil dari penilaian kuesioner yang telah diisi oleh pengguna dalam pengujian usabilitas.





Tabel 4.4 Hasil pengujian usabilitas

No	Indikator	Pertanyaan	Nilai	Keterangan	Rata-rata	Keterangan
1	Lea	Apakah aplikasi memiliki tampilan yang mudah dipahami dan mudah digunakan?	4,6	Sangat Layak	4,4	Layak
2		Apakah aplikasi menggunakan bahasa percakapan yang mudah dipahami?	4,3	Layak		
3	Efe	Apakah aplikasi mampu memahami masukan pengguna dalam melakukan permintaan penjemputan sampah?	4,1	Layak	4,3	Layak
4		Apakah aplikasi mampu memberikan respons yang sesuai dengan yang diharapkan?	4,5	Layak		
5	MeR	Apakah aplikasi tidak membingungkan saat digunakan?	4,1	Layak	4,2	Layak
6		Apakah aplikasi minim memberikan kesalahan saat merespons permintaan pengguna?	4,3	Layak		
7	Efi	Apakah aplikasi dapat memberikan manfaat kepada pengguna dalam melakukan permintaan penjemputan sampah?	4,4	Layak	4,4	Layak
8		Apakah aplikasi dapat memproses permintaan penjemputan sampah yang dilakukan oleh pengguna?	4,5	Layak		
9	Sat	Apakah dengan adanya aplikasi ini untuk melakukan penjemputan sampah dapat memberikan kemudahan kepada pengguna dalam pengelolaan sampah?	4,5	Layak	4,4	Layak
10		Apakah pengguna puas dengan adanya aplikasi ini untuk melakukan permintaan jemput sampah?	4,3	Layak		
Skor Usabilitas					4,36	Layak

Berdasarkan nilai data yang diperoleh dari setiap pertanyaan, maka didapatkan nilai rata-rata 4,36 dari nilai maksimal 5,0. Dari nilai rata-rata yang diperoleh, tingkat usability pada aplikasi yang diujikan berada pada kategori layak yang artinya aplikasi sudah memenuhi kelayakan untuk digunakan oleh pengguna.



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah melakukan tahapan demi tahapan mulai dari studi literatur sampai pengujian, dapat ditarik kesimpulan bahwa pengembangan aplikasi sistem jemput sampah untuk rumah kos dan area kampus berbasis Android telah berhasil dilakukan. Aplikasi yang dikembangkan dapat menghubungkan antara masyarakat dengan petugas untuk melakukan permintaan jemput sampah dan menjemput sampah oleh petugas. Masyarakat dan mahasiswa khususnya yang berada di rumah kos atau di area kampus yang memiliki kebiasaan membuang sampah secara sembarangan bisa melakukan permintaan penjemputan sampah kepada petugas secara *on-demand*. Dengan memanfaatkan *Mapbox Maps SDK* dan *Mapbox Navigation SDK* petugas bisa mengetahui titik lokasi penjemputan sampah yang dikirimkan masyarakat dan mahasiswa.

#### 5.2 Saran

Aplikasi yang telah dikembangkan pada penelitian ini masih memiliki beberapa kekurangan yang bisa dikembangkan lagi diantaranya menambahkan integrasi ke berbagai layanan pembayaran digital dan juga menambahkan fitur tukar poin untuk bisa ditukarkan pada aplikasi. Karena pada penelitian ini aplikasi yang dikembangkan hanya berjalan di *platform* Android, maka penelitian selanjutnya dapat melanjutkan pengembangan aplikasi ini untuk platform lain seperti iOS supaya cakupan penggunaanya lebih luas. Dapat juga diteliti penerimaan masyarakat akan aplikasi ini sehingga dapat diukur dampak nyatanya, baik dalam hal pembuangan dan pengelolaan sampah maupun kebersihan lingkungan secara umum.

## DAFTAR PUSTAKA

- Caca, R. (2020). *Pemerintah: Sudah Pedulikan Terhadap Masalah Sampah Plastik?* Retrieved May 13, 2020, from <https://www.suara.com/yoursay/2020/01/16/142453/pemerintah-sudah-pedulikan-terhadap-masalah-sampah-plastik>
- Katadata, T. P. (2019). *Menuju Indonesia Peduli Sampah - Infografik Katadata.co.id.* <https://katadata.co.id/infografik/2019/12/10/menuju-indonesia-peduli-sampah>
- Kamus Istilah Lingkungan untuk Manajemen, Ecolink, 1996.
- Nur Latifah Jamaluddin. (2016). *ANALISIS DAN PERANCANGAN SISTEM PENGELOLAAN SAMPAH BERBASIS ANDROID MELALUI PARTISIPASI MASYARAKAT DI KECAMATAN RAPPOCINI.* 1–10.
- Ridho Firmansya, Nurhadi, M. (2019). Perancangan aplikasi sig lokasi bank sampah di kota jambi berbasis android. *Jurnal Ilmiah Mahasiswa Teknik Informatika*, 1(4).
- Aziz, A., Fajar, S., & Gumilang, S. (2018). Rancangan Fitur Aplikasi Pengelolaan Administrasi dan Bisnis Bank Sampah di Indonesia. *Konferensi Nasional Sistem Informasi*, 208–213.
- Masrurroh, S. U., Suciasih, S. E., & Suseno, H. B. (2015). Pengembangan Aplikasi Bank Sampah Menggunakan Layanan Teknologi Informasi Cloud Computing Pada Bank Sampah Melati Bersih. *Jurnal Teknik Informatika*, 8(2), 1–9. <https://doi.org/10.15408/jti.v8i2.2403>
- Handarkho, Y. D., & Irianto, A. B. P. (2016). Implementasi Sistem Informasi Manajemen Bank Sampah Untuk Meningkatkan Kinerja Usaha Kecil Menengah (Studi Kasus Bank Sampah Gemah Ripah Badegan, Bantul). *Telenomatika*, 9(1), 2–13.
- Magdalena, H., Santoso, H., & Rochmayani, K. (2019). Sistem Retribusi Sampah Berbasis Web untuk Optimalisasi Kinerja Bidang Pengelolaan Sampah Web-Based Waste Retribution Information System for Optimizing Performance in Waste Management. *Cogito Smart Journal*, 5(2), 294–307.
- Kumurur, V. (2012). *Pengetahuan, Sikap Dan Kepedulian Mahasiswa Pascasarjana Ilmu Lingkungan Terhadap Lingkungan Hidup Kota Jakarta.* 8(2), 1–24.
- Mobile Operating System Market Share Indonesia | StatCounter Global Stats.* (n.d). Retrieved March 31, 2021, from <https://gs.statcounter.com/os-market-share/mobile/indonesia>
- Number of monthly Android app releases worldwide 2021 | Statista.* (n.d.). Retrieved March 31, 2021, from <https://www.statista.com/statistics/1020956/android-app-releases-worldwide/>
- Fused Location Provider API | Google Developers.* (n.d.). Retrieved March 31, 2021, from <https://developers.google.com/location-context/fused-location-provider>

*Geocoder | Developer Android | Android Developers.* (n.d.). Retrieved March 31, 2021, from <https://developer.android.com/reference/android/location/Geocoder>

*Maps | Mapbox.* (n.d.). Retrieved November 18, 2020, from <https://www.mapbox.com/maps/>

Bulatovych, D. (n.d.). *Mapbox vs Google Maps Platform: Which Tool Is Better for Your Logistics App.* Retrieved April 1, 2021, from <https://yalantis.com/blog/mapbox-maps-ready-mobile-apps/>

*Navigation SDK | Android | Mapbox.* (n.d.). Retrieved November 18, 2020, from <https://docs.mapbox.com/android/navigation/overview/>

*Firebase Realtime Database.* (n.d.). Retrieved April 2, 2021, from <https://firebase.google.com/docs/database>

*Firebase Cloud Messaging.* (n.d.). Retrieved April 2, 2021, from <https://firebase.google.com/docs/cloud-messaging>

*Cloud Storage for Firebase.* (n.d.). Retrieved April 2, 2021, from <https://firebase.google.com/docs/storage>



LAMPIRAN

