



**APLIKASI ANDROID UNTUK PELAPORAN  
PENGADUAN MASYARAKAT UNTUK  
TEMPAT PEMBUANGAN SAMPAH**

Disusun Dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir  
Memperoleh Gelar Strata 1 Pada Fakultas Sains Dan Teknologi  
Universitas Pembangunan Panca Budi  
Medan

**SKRIPSI**

**OLEH**

**NAMA : M. GUNTUR A. TANJUNG**  
**N.P.M : 1724370833**  
**PROGRAM STUDI : SISTEM KOMPUTER**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PEMBANGUNAN PANCA BUDI**

**MEBAN**

**2019**

## **ABSTRAK**

**M. GUNTUR A. TANJUNG**

**Aplikasi *Android* Pelaporan Pengaduan Masyarakat Untuk Tempat  
Pembuangan Sampah  
2019**

Produksi sampah setiap hari semakin meningkat seiring dengan bertambahnya jumlah produk dan pola konsumsi masyarakat. Hal yang harus dilakukan untuk mengatasi peningkatan volume sampah tersebut adalah dengan cara: mengurangi volume sampah dari sumbernya melalui pemberdayaan masyarakat. Permasalahan dalam partisipasi masyarakat mengenai pengelolaan sampah adalah apa saja bentuk regulasi yang terkait dengan pengelolaan sampah. Untuk mencapai kondisi masyarakat yang hidup sehat dan sejahtera di masa yang akan datang, akan sangat diperlukan adanya lingkungan permukiman yang sehat. Dari aspek persampahan, maka kata sehat akan berarti sebagai kondisi yang akan dapat dicapai bila sampah dapat dikelola secara baik sehingga bersih dari lingkungan permukiman dimana manusia beraktifitas di dalamnya. Persoalan lingkungan yang selalu menjadi isu besar di hampir seluruh wilayah perkotaan adalah masalah sampah. *Android* merupakan sebuah sistem operasi yang sangat di gemari masyarakat belakangan ini. *Android* dan sampah adalah dua hal yang jauh berbeda dalam sebuah pengertian. Tetapi jika dapat disatukan menjadi sebuah sistem dapat mempermudah masyarakat dalam mengelola sampah. Sebuah sistem aplikasi berbasis *android* untuk pengaduan-pengaduan sampah yang menumpuk.

**Kata Kunci: *Android, Aplikasi, Sampah***



## DAFTAR ISI

	<b>Halaman</b>
<b>KATA PENGANTAR .....</b>	<b>i</b>
<b>DAFTAR ISI .....</b>	<b>iii</b>
<b>DAFTAR GAMBAR .....</b>	<b>vi</b>
<b>DAFTAR TABEL .....</b>	<b>viii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>ix</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
<b>BAB II LANDASAN TEORI .....</b>	<b>4</b>
2.1 Pengertian Aplikasi .....	4
2.2 Pengertian <i>Android</i> .....	5
2.1.1 Sejarah <i>Android</i> .....	6
2.1.2 Arsitektur <i>Android</i> .....	9
2.1.3 Kelebihan Dan Kekurangan Operasi Sistem <i>Android</i> .....	9
2.1.4 <i>Android Studio</i> .....	10
2.2 UML ( <i>Unified Modelling Language</i> ).....	11
2.2.1 UML ( <i>Unified Modelling Language</i> ).....	11
2.2.2 <i>Activity Diagram</i> .....	15
2.2.3 <i>Sequence Diagram</i> .....	15
2.2.4 <i>State Diagram</i> .....	16
2.4 <i>DataBase</i> .....	16
2.4.1 Kegunaan <i>DataBase</i> .....	16
2.4.2 <i>SQLite</i> .....	17
2.4.3 Fitur <i>SQLite</i> .....	18
2.5 Sampah .....	23
2.6 <i>Kotlin</i> .....	24
2.7 <i>Google Map API</i> .....	26

2.8	<i>Back4app</i> .....	26
<b>BAB III METODE PENELITIAN .....</b>		<b>27</b>
3.1	Tahapan Penelitian .....	27
3.2	Metode Pengumpulan Data .....	29
3.3	Analisis Sistem Sedang Berjalan .....	30
3.3.1	Identifikasi Sistem sedang berjalan .....	30
3.3.2	Kelemahan Sistem yang Berjalan .....	30
3.3.3	Gambaran Umum Sistem yang Diusukan .....	31
3.4	Rancangan Penelitian .....	31
3.4.1	<i>Use Case Diagram</i> .....	31
3.4.2	<i>Activity Diagram</i> .....	33
3.4.3	<i>Activity Diagram Admin</i> .....	34
3.4.4	<i>Activity Diagram User</i> .....	35
3.4.5	<i>Activity Diagram Login Admin</i> .....	36
3.4.6	<i>Activity Diagram Daftar User</i> .....	37
3.4.7	<i>Activity Diagram Login User</i> .....	38
3.4.8	<i>Activity Diagram Laporan Baru</i> .....	39
3.4.9	<i>Sequence Diagram</i> .....	40
3.5	<i>Sequence Diagram Admin</i> .....	40
3.6	<i>Sequence Diagram Login Admin</i> .....	41
3.7	<i>Sequence Diagram User</i> .....	41
3.8	<i>Sequence Diagram Login User</i> .....	42
3.10	Perancangan Aplikasi <i>Android</i> .....	43
3.6.1	Tampilan Awal Aplikasi <i>Admin</i> .....	43
3.6.2	Tampilan Menu Baru di Aplikasi <i>Admin</i> .....	44
3.6.3	Tampilan Menu Proses di Aplikasi <i>Admin</i> .....	44
3.6.4	Tampilan Menu Selesai di Aplikasi <i>Admin</i> .....	45
3.6.5	Tampilan Menu Awal Aplikasi <i>User</i> .....	46
3.6.6	Tampilan Menu Form di Menu Daftar .....	47
3.6.7	Tampilan Menu Pesan .....	48
3.6.8	Tampilan Menu Akun .....	49
3.6.9	Tampilan Ganti Password .....	50
3.6.10	Tampilan Edit Profile .....	51
3.6.11	Tampilan Edit Foto Profile .....	52

<b>BAB IV HASIL PENELITIAN DAN PEMBAHASAN .....</b>	<b>53</b>
4.1 Hasil Tampilan Pada Sistem Admin .....	53
4.1.1 Tampilan Halaman Utama di Sistem Admin .....	53
4.1.2 Tampilan Halaman Laporan Baru .....	53
4.1.3 Tampilan Halaman Proses.....	54
4.1.4 Tampilan Halaman Proses.....	54
4.2 Hasil Tampilan Pada Sistem <i>User</i> .....	55
4.2.1 Tampilan Halaman Utama di Sistem Admin . .....	55
4.2.2 Tampilan HalamanUtama Daftar User .....	56
4.2.3 Tampilan Halaman Laporan/Riwayat Laporan.....	56
4.2.4 Tampilan Halaman Kirim Laporan Baru . .....	57
4.2.5 Tampilan Halaman Pesan .....	57
4.2.6 Tampilan Akun User .....	58
<b>BAB V PENUTUP .....</b>	<b>60</b>
5.1 Kesimpulan .....	60
5.2 Saran.....	60
<b>DAFTAR PUSTAKA .....</b>	<b>61</b>
<b>BIOGRAFI PENULIS</b>	
<b>LAMPIRAN-LAMPIRAN</b>	

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 <i>Logo Android</i> .....	8
Gambar 3.1 <i>Flowchart Tahapan Penelitian</i> .....	27
Gambar 3.2 <i>Use Case Diagram Admin</i> .....	32
Gambar 3.3 <i>Use case Diagram User</i> .....	33
Gambar 3.4 <i>Activity Diagram Admin</i> .....	34
Gambar 3.5 <i>Activity Diagram Login User</i> .....	35
Gambar 3.6 <i>Activity Diagram Login Admin</i> .....	36
Gambar 3.7 <i>Activity Diagram Daftar User</i> .....	37
Gambar 3.8 <i>Activity Diagram Login User</i> .....	38
Gambar 3.9 <i>Activity Diagram Laporan Baru</i> .....	39
Gambar 3.10 <i>Sequence Diagram Admin</i> .....	40
Gambar 3.11 <i>Sequence Diagram Login Admin</i> .....	41
Gambar 3.12 <i>Sequence Diagram User</i> .....	41
Gambar 3.13 <i>Sequence Diagram Login User</i> .....	42
Gambar 3.14 <i>Tampilan awal aplikasi admin</i> .....	43
Gambar 3.15 <i>Tampilan awal aplikasi admin</i> .....	44
Gambar 3.16 <i>Tampilan menu Proses</i> .....	44
Gambar 3.17 <i>Tampilan menu Selesai</i> .....	45
Gambar 3.18 <i>Tampilan awal aplikasi User</i> .....	46
Gambar 3.19 <i>Tampilan Form di menu Daftar</i> .....	47
Gambar 3.20 <i>Tampilan menu Pesan</i> .....	48
Gambar 3.21 <i>Tampilan Menu Akun</i> .....	49
Gambar 3.22 <i>Tampilan Ganti Password</i> .....	50
Gambar 3.23 <i>Edit Profile</i> .....	51

Gambar 3.24 <i>Edit Foto Profile</i> .....	52
Gambar 4.1 Tampilan <i>login admin</i> .....	53
Gambar 4.2 Tampilan halaman laporan baru <i>admin</i> .....	54
Gambar 4.3 Tampilan halaman Laporan Baru <i>admin</i> .....	54
Gambar 4.4 Tampilan halaman selesai <i>admin</i> .....	55
Gambar 4.5 Tampilan halaman login <i>user</i> .....	55
Gambar 4.6 Tampilan halaman utama daftar <i>user</i> .....	56
Gambar 4.7 Tampilan halaman laporan/riwayat laporan <i>user</i> .....	56
Gambar 4.8 Tampilan halaman form kirim laporan baru <i>user</i> .....	57
Gambar 4.9 Tampilan halaman pesan <i>user</i> .....	57
Gambar 4.10 Tampilan halaman akun <i>user</i> .....	58
Gambar 4.11 Tampilan <i>edit profile</i> .....	58
Gambar 4.12 Tampilan merubah <i>password user</i> .....	59



## DAFTAR TABEL

	Halaman
Tabel 2.1 Elemen <i>Use Case Diagram</i> .....	14
Tabel 2.2 Elemen-Elemen <i>Activity Diagram</i> .....	16
Tabel 2.3 Elemen-Elemen <i>Sequence Diagram</i> .....	19
Tabel 2.4 Simbol-simbol <i>State Diagram</i> .....	20

## DAFTAR LAMPIRAN

	Halaman
List Program.....	62

## KATA PENGANTAR



*Assalamu'alaikum Warahmatullahi Wabarakatu,*

Syukur Alhamdulillah penulis ucapkan kepada Allah SWT Tuhan semesta alam yang telah memberikan kemudahan, pertolongan, kasih sayang serta anugerah yang tak terhingga kepada penulis sehingga penyusunan Skripsi yang berjudul “**Aplikasi Android Pelaporan Pengaduan Masyarakat Untuk Tempat Pembuangan Sampah**” dapat penulis selesaikan dengan baik. Shalawat beserta salam tak lupa penulis ucapkan kepada Nabi besar Muhammad SAW, yang telah menghantarkan umatnya dari alam kegelapan dan kebodohan menuju alam yang terang benderang dan penuh dengan ilmu pengetahuan, serta telah memberikan contoh akhlakul karimah bagi seluruh muslim di seluruh penjuru dunia.

Penulis menyadari bahwa dalam penulisan Skripsi ini tidak terlepas bantuan dan dukungan dari berbagai pihak, baik secara moril maupun materi. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih dan penghargaan sedalam-dalamnya kepada :

- 1) Muhammad DR. H. Muhammad Isa Indrawan, SE, M.M, selaku Rektor Universitas Pembangunan Panca Budi Medan.
- 2) Ir. Bhakti Alamsyah, M.T, Ph.D, selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
- 3) Eko Hariyanto, S.Kom, M.Kom, selaku Ketua Program Studi Sistem Komputer di Universitas Pembangunan Panca Budi Medan.

- 4) Dr. Muhammad Iqbal, S.Kom., M.Kom selaku dosen pembimbing I dan Barany Fachri, ST., M. Kom, selaku dosen pembimbing II penulis yang telah banyak memberikan saran dan bimbingan, serta pengarahan kepada penulis hingga dapat menyelesaikan Skripsi ini.
- 5) Seluruh Dosen-Dosen Universitas Pembangunan Panca Budi Medan yang telah mengajarkan banyak hal kepada penulis.
- 6) Teristimewa kepada kedua orang tua yang penulis sayangi Ayahanda Adaan Tanjung, Ibunda Maryam Siregar, serta saudara- saudara yang tidak bisa penulis sebutkan satu-persatu yang tiada henti-hentinya mendoakan dan mendukung penulis, agar kelak menjadi orang yang berguna bagi agama, bangsa dan negara.
- 7) Teman-teman seperjuangan yang selalu memberikan motivasi dan bantuan dalam menyelesaikan tugas akhir ini yang memberikan semangat dan dorongan dalam penyelesaian skripsi ini khususnya kelas karyawan Sistem Komputer.

Penulis menyadari bahwa dalam penulisan Skripsi ini masih terdapat banyak kekurangan baik dalam sistematika penulisan maupun penggunaan bahasa yang tidak sesuai dengan aturan yang ditetapkan. Untuk itu penulis menerima kritik dan saran yang dapat membangun dalam penyempurnaan Skripsi ini, agar lebih bermanfaat dikemudian hari. Semoga penulisan Skripsi ini dapat bermanfaat bagi semua pihak yang membacanya. Akhir kata, penulis mengucapkan terima kasih.

Medan, Sep 2019

Penulis

(M. GUNTUR A. TANJUNG)

NPM : 1724370833

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Banyak masyarakat yang tidak peduli dengan kebersihan lingkungan sekitar rumah. Salah satunya adalah masih banyaknya masyarakat dalam membuang sampah secara sembarangan. Sampah adalah suatu barang atau benda yang sudah tidak berguna terbuang dari kegiatan sehari-hari. Sampah yang terkumpul dapat menumpuk dan membusuk sehingga sangat mengganggu kesehatan lingkungan serta memengaruhi mutu estetika. Sampah-sampah itupun ada yang mudah terurai dan tidak, bahkan ada yang memerlukan waktu hingga 100 tahun agar bisa terurai. Hal itulah yang menyebabkan sampah terus menumpuk yang tentunya bisa berakibat merugikan bagi masyarakat.

Di suatu kota tertentu sampah masih saja menjadi masalah yang tak ada habisnya. Tumpukan-tumpukan sampah kembali terlihat di beberapa ruas jalan dan tempat pembuangan sampah dalam kota. Masyarakat mengeluh sampah menumpuk dan mengeluarkan bau tak sedap. Untuk menghindari masalah penumpukan sampah, maka aplikasi untuk pelaporan pengaduan masyarakat untuk pembuangan sampah merupakan suatu upaya dalam memberikan informasi mengenai tempat pembuangan sampah yang sudah menumpuk atau belum di angkut oleh petugas kebersihan, sehingga dengan adanya informasi pengaduan ini, diharapkan sampah-sampah tidak menumpuk pada suatu tempat dalam waktu yang lama.

Aplikasi ini juga untuk melengkapi dari penelitian sebelumnya yang berjudul “Prototipe Aplikasi Layanan Masyarakat Berbasis *Android* dan *Web Service*” (Andi Jumardi, 2016). Aplikasi tersebut dirancang dengan menggunakan *web service* sebagai suatu fasilitas untuk menyediakan layanan dalam bentuk informasi kepada sistem lain dan *Location Based Services* untuk menentukan posisi atau lokasi dimana user melakukan pengaduan. Dengan melengkapi dari penelitian sebelumnya, penelitian ini akan lebih memudahkan masyarakat untuk mengelola dan melaporkan masalah terkait pembuangan sampah kepada dinas kebersihan kota, seperti: sampah sudah penuh dan tidak kunjung diambil oleh petugas dinas kebersihan. Tujuan penelitian ini adalah agar lingkungan kota menjadi bersih dan sehat.

## **1.2 Perumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan di atas, maka perumusan masalah dalam perancangan aplikasi *android* pengaduan masyarakat untuk pembuangan sampah ini adalah:

1. Bagaimana cara merancang aplikasi *android* pengaduan masyarakat untuk pembuangan sampah ?
2. Bagaimana cara merancang tampilan aplikasi agar mudah dimengerti pengguna ?

## **1.3 Batasan Masalah**

Agar penelitian lebih terarah dan fokus terhadap tujuan, maka penelitian ini dibatasi dengan batasan masalah, sebagai berikut:

1. Ruang lingkup aplikasi yang dibuat hanya menangani pengaduan masyarakat untuk pembuangan sampah.
2. Aplikasi yang dibuat wilayah pemetaan hanya pada Medan dan sekitarnya.
3. *Database* yang digunakan pada aplikasi ini menggunakan *platform Back4app*.

#### **1.4 Tujuan Penelitian**

Adapun tujuan dari proyek akhir ini adalah:

1. Merancang aplikasi *android* untuk pengaduan masyarakat untuk pembuangan sampah yang menumpuk.
2. Merancang tampilan aplikasi agar mudah dimengerti pengguna.
3. Membangun aplikasi *android* yang dapat mengolah data rekapitulasi sampah yang telah diangkut atau belum diangkut

#### **1.5 Manfaat Penelitian**

Adapun manfaat dari penelitian ini adalah membantu masyarakat untuk melakukan pengaduan terhadap dinas kebersihan mengenai tempat pembuangan sampah, seperti: sampah sudah penuh dan tidak kunjung diambil oleh petugas Dinas Kebersihan, sehingga dengan adanya informasi pengaduan ini, diharapkan sampah-sampah tidak menumpuk pada suatu tempat dalam waktu yang lama.

## BAB II

### LANDASAN TEORI

#### 2.1 Pengertian Aplikasi

Dikutip dari situs (pengajar.co.id : 2019), aplikasi merupakan program yang siap untuk dipakai, dibuat untuk melaksanakan perintah pemakainya. Arti lain dari aplikasi ialah kumpulan program yang dibuat untuk melakukan pekerjaan khusus yang di perintahkan oleh pemakainya. Atau bisa juga dikatakan aplikasi adalah perangkat lunak (*software*) yang di buat untuk mempermudah pekerjaan manusia. Selain bisa membantu dan mempercepat proses pekerjaan manusia, aplikasi juga bisa menciptakan hasil yang lebih akurat dalam memecahkan permasalahan.

Dan beberapa pengertian aplikasi menurut para ahli, menurut Harip Santoso, aplikasi merupakan sebuah kelompok *file* (*form, class, report*) yang bertujuan untuk melakukan kegiatan tertentu yang saling terhubung, contohnya seperti aplikasi *payroll*, aplikasi *fixed asset*, dll. Menurut Ibis, aplikasi merupakan alat bantu untuk mempermudah dan juga mempercepat proses pekerjaan serta bukan merupakan beban bagi pemakainya. Dan menurut Hengky W. Pramana, aplikasi adalah sebuah unit perangkat lunak yang dibuat untuk melayani keperluan akan beberapa aktivitas, contohnya seperti sistem perniagaan, game pelayanan masyarakat, periklanan maupun semua proses yang hampir dilakukan manusia.



## 2.2 Pengertian *Android*

Menurut Faris, dkk (2017), *Android* merupakan subset perangkat lunak untuk perangkat mobile yang meliputi sistem operasi middleware dan aplikasi inti yang di *release* oleh Google. Salah satu pencipta dari *Android* adalah *Andy Rubin*, yang kini sering disebut sebagai “Bapak *Android*”. Pada tahun 2005, *Google* secara resmi telah membeli *Android*. Sehingga sejak saat itu, pengembangan *Android* sepenuhnya berada di tangan *Google* hingga saat ini. Namun *Google* tetap merilis kode sumber (*source code*) secara terbuka, sehingga *Android* termasuk dalam *software open source*. Yang artinya, setiap orang di seluruh dunia juga dapat berkontribusi untuk mengembangkan *Android*. Jadi jika kita simpulkan, pengertian *Android* menurut para ahli adalah sebuah sistem operasi yang dikembangkan khusus untuk perangkat *smartphone* dan *tablet*. Hingga kini *Android* masih terus dikembangkan dan berikut ini adalah versi-versi *Android* dari yang paling lama hingga yang terbaru (Rizki, 2019) :

- *Android Version 1.0*

Diluncurkan pertama kali pada september 2008, meskipun setahun sebelumnya *Google* telah merilis versi beta yang diperkenalkan kepada khalayak ramai. *Android* versi pertama dikenal dengan *android 1.0 (Android One)*. Saat itu *Google* belum menamai *android* ini dengan nama pencuci mulut.

- *Android Version 1.5 (Cupcake)*

Pada April 2009 *Google* menggunakan nama cemilan untuk yang pertama kalinya pada versi *android*.

- *Android Version 1.6 (Donut)*  
*Android* melanjutkan penamaan dengan nama kue Donat, *Donut* 1.6 ini sendiri dirilis pada 15 September 2009.
- *Android Version 2.0/2.1 (Eclair)*  
Dirilis pada tanggal 9 Desember 2009.
- *Android Version 2.2 (Frozen Yougurt/Froyo)*  
Lahir pada 20 Mei 2010. *Android* ini mulai dilengkapi dengan fitur *friendly user* seperti opsi untuk mematikan akses data pada jaringan seluler.
- *Android Version 2.3 (Gingerbread)*  
Diperkenalkan pada tanggal 6 Desember 2010, fitur andalan dari *android* ini adalah *Download Manager* yang memungkinkan pengguna untuk mengunduh *file* secara langsung dari internet. Juga *men-support* multi kamera.
- *Android Version 3.0/3.1/3.2 (Honeycomb)*  
Diliris pada 22 Februari 2011, kemudian pada Mei 2011 versi 3.1 diperkenalkan ke publik. Memiliki fitur unggulan dalam perbaikan *bug* dan keamanan, kestabilan, dan peningkatan kinerja *Wi-Fi* pada OS *android*.
- *Android Version 4.0/4.0.4 (Ice Cream Sandwich)*  
Dirilis pada 19 Oktober 2011, penambahan buka kunci dengan identifikasi wajah pertama kali diusung pada versi ini.

- *Android Version 4.1/4.2 (Jelly Bean)*

Dirilis pada 9 Juli 2012 sudah didapatkan fitur untuk mematikan notifikasi yang dianggap mengganggu pada aplikasi tertentu.

- *Android Version 4.4 (Kitkat)*

*Android* ini dirilis pada 31 Oktober 2013 dan menjadi salah satu versi *android* yang paling disukai oleh pengguna *smartphone* didunia, karena *kitkat* memiliki fitur dengan optimasi kinerja untuk *smartphone* dengan spesifikasi rendah.

- *Android Version 5.0 (Lolipop)*

25 Juni 2014 diperkenalkan pertama kali, adanya penamaan fitur “*Factory Reset Protection*” fitur ini berguna untuk perlindungan *factory Reset* dimana pengguna akan diminta *password* ketika *android* di *reset*.

- *Android Version 6.0 (Marshmallow)*

Sistem proteksi *android* dengan sidik jari pertama sekali dirancang pada OS *android Marshmallow*.

- *Android Version 7.0 (Nougat)*

*Android* versi ini dilengkapi dengan mode *Multitasking* yang membuat para pengguna bisa menjalankan 2 aplikasi dalam 1 layar, selain itu kecepatan untuk *moving* dari satu layar ke layar lainnya sangat *smooth* dan *responsive* bagi para pengguna.

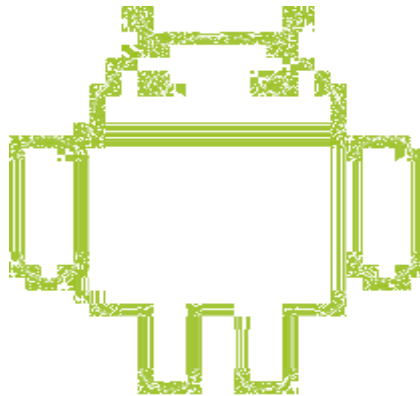
- *Android Version 8.0 (Oreo)*

Desember 2017 *android* versi ini resmi dirilis dengan peningkatan fitur-fitur yang lebih canggih.

- *Android Version 9.0 (Pie)*

Sistem operasi *Android* versi kesembilan itu mulai tersedia lewat *update over-the-air* untuk perangkat-perangkat seri *Pixel* besutan *Google*.

Jika diperhatikan, semua nama versi *Android* diatas menggunakan nama nama makanan penutup. Dan secara berurutan secara *alfabet*. Ini memang sudah jadi “tradisi” bagi *Google* dalam pemberian nama versi untuk *Android*.



**Gambar 2.1** Logo *Android*

### **2.2.1 Sejarah *Android***

Sebelum dimiliki oleh *Google*, *Android* sendiri adalah sistem operasi yang dikembangkan oleh perusahaan bernama *Android, Inc.* yang didirikan di *Palo Alto, California* pada bulan Oktober 2003 oleh *Andy Rubin, Rich Miner, Nick Sears, dan Chris White*. Perusahaan ini sendiri memang sedari awal sudah didukung secara finansial oleh *Google* sebelum akhirnya dibeli pada tahun 2005. Walaupun sudah dibeli oleh *Google*, tapi para pendiri *Android*, seperti *Rubin, Miner dan White* tetap bekerja di perusahaan (Rizki, 2019).

### 2.2.2 Arsitektur *Android*

Secara umum arsitektur *android* terdiri dari *application*, *application framework*, *libraries*, *android runtime* dan *linux kernel*. Berikut ini penjelasan dari arsitektur android (Rizki, 2019) :

#### 1) *Application*

*Application* adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita mengunduh aplikasi kemudian kita melakukan instalasi dan menjalankan aplikasi tersebut. Di layer ini memiliki aplikasi inti termasuk *email*, *SMS*, kalender, peta, *browser*, kontak, dan lain-lain. Keseluruhan aplikasi ditulis menggunakan bahasa pemrograman *Java*.

#### 2) *Application Framework*

*Applications framework* adalah layer dimana pembuat aplikasi melakukan pembangunan aplikasi yang akan dijalankan di sistem operasi *android*, karena pada layer inilah aplikasi dapat dibangun dan dirancang, seperti *content-providers* yang berupa sms dan panggilan telepon.

#### 3) *Libraries*

*Libraries* adalah layer dimana seluruh fitur *android* berada, biasanya para pengembang aplikasi mengakses *libraries* untuk menjalankan aplikasinya.

#### 4) *Android Runtime*

Layer ini merupakan layer yang memungkinkan aplikasi *android* bisa dijalankan dalam prosesnya menggunakan implementasi *Linux*.

#### 5) *Dalvik Virtual machine*

(DVM) adalah mesin yang membentuk dasar kerangka pada aplikasi *android*.

#### 6) *Linux Kernel*

*Linux Kernel* adalah layer inti dari suatu sistem *android*. Layer ini berisikan *file-file sistem* yang mengatur sistem *processing*, *memori*, *resource*, *driver*, dan sistem-sistem *android* yang lainnya.

### 2.2.3 Kelebihan Dan Kekurangan Operasi Sistem *Android*

Beberapa keunggulan *android* dibandingkan dengan bahasa pemrograman lain ialah sebagai berikut (Andriani, 2016):

#### 1) Kelebihan *Android*

##### a. *User Friendly*

Sistem *android* sangat mudah untuk dijalankan. Sama halnya dengan sistem operasi *windows* yang ada pada komputer.

##### b. Kemudahan Dalam Notifikasi

Setiap ada *SMS*, *Email*, atau bahkan artikel terbaru dari *RSS Reader*, maka akan selalu ada notifikasi di *Home Screen* ponsel *android*.

##### c. Aplikasi

Aplikasi akan disajikan jutaan pilihan aplikasi yang menarik dari yang gratis hingga yang berbayar, dan anda bisa mendownloadnya di *Google Play*.

##### d. *Multitasking*

*Android* bisa menjalankan berbagai aplikasi , yang artinya bisa *browsing* sambil mendengarkan lagu.

## 2) Kekurangan *Android*

### a. Koneksi *Internet* Yang Terus Menerus

Kebanyakan ponsel *android* memerlukan koneksi *internet* yang terus menerus aktif. Hal ini terjadi jika kita menyalakan paket data dan menggunakan *widget* serta aplikasi yang berjalan secara berlebihan.

### b. *Update System* Yang Kurang Efektif

Untuk mengupdate sistem *android* bukanlah mudah. Kita diharuskan untuk menunggu masing-masing vendor merilis resmi *update* terbaru dari sistem *android* tersebut.

### c. Iklan

Konsekuensi di setiap aplikasi akan selalu ada iklan yang terpampang entah itu dibagian atas atau bawah aplikasi.

## 2.2.4 **Android Studio**

*Android Studio* adalah *Integrated Development Enviroment* (IDE) untuk sistem operasi *Android*, yang dibangun diatas perangkat lunak *JetBrains IntelliJ IDEA* dan didesain khusus untuk pengembangan *Android*. IDE ini merupakan pengganti dari *Eclipse Android Development Tools* (ADT) yang sebelumnya merupakan IDE utama untuk pengembangan aplikasi *android* (Maiyana:2018).

*Android studio* sendiri pertama kali diumumkan di *Google I/O conference* pada tanggal 16 Mei 2013. Ini merupakan tahap preview dari versi 0.1 pada Mei 2013, dan memasuki tahap beta sejak versi 0.8 dan mulai diliris pada Juni 2014.

Versi liris stabil yang pertama diliris pada Desember 2014, dimulai sejak versi 1.0. Sedangkan versi stabil yang sekarang adalah versi 3.13 yang diliris pada Juni 2018. Fitur yang tersedia saat ini dalam *stable version*.

- Dukungan *Gradle-based build*
- *Android-specific refactoring* dan perbaikan cepat
- *Lint tools* untuk menangkap kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya
- *Integrasi Proguard* dan kemampuan penananda tangan aplikasi
- *Template-based wizards* untuk membuat *template design* umum seperti *drawer* atau *empty activity*
- Mendukung untuk pengembangan aplikasi *Android Wear*.
- Editor tata letak yang memungkinkan pengguna untuk menyeret dan menjatuhkan (drag-and-drop) komponen UI, opsi untuk melihat tata letak pada beberapa konfigurasi layar
- Dukungan bawaan untuk *Google Cloud Platform*, memungkinkan integrasi dengan *Firebase Cloud Messaging* ('Perpesanan *Google Cloud*' Sebelumnya) dan *Google App Engine*
- *Android Virtual Device* (Emulator) untuk menjalankan dan men-debug aplikasi di studio *Android*.

### 2.3 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan,



membangun, dan pendokumentasian dari sebuah sistem pengembangan *Software* berbasis *Object Oriented*. UML juga memberikan *standart* penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (Rahmanto:2017).

UML adalah salah satu *tools/model* untuk merancang pengembangan *software* yang berbasis *object oriented*. UML sebagai sebuah bahasa yang memberikan *vocabulary* dan tatanan penulisan kata-kata dalam *Microsoft Word* untuk kegunaan komunikasi.

UML tidak hanya merupakan sebuah bahasa pemrograman *visual* saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti *Java*, *C++*, *Visual Basic*, atau bahkan dihubungkan secara langsung ke dalam sebuah *object oriented database*.

### **2.3.1 UML (*Unified Modelling Language*)**

UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML merupakan singkatan dari *Unified Modeling Language*. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena *developer* harus melakukan penelusuran dan mempelajari kode program(Rahmanto:2017).

UML juga dapat menjadi alat bantu untuk *transfer* ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainnya.

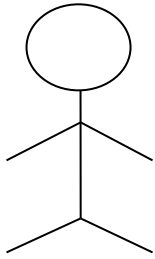
Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya UML.

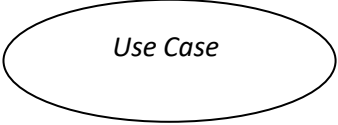


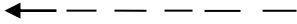
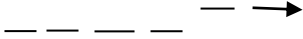
UML diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. Dalam pengembangan berorientasi objek ada beberapa prinsip yang harus dikenal:


- *Object*
- *Class*
- *Abstraction*
- *Encapsulation*
- *Inheritance*
- *Polymorphism*

Notasi yang digunakan dalam *use case* diagram adalah:

**(Tabel 2.1 : Elemen *Use Case Diagram*)**

Nama Elemen Dan Fungsi	Notasi
<p><i>Actor</i> : Menggambarkan tokoh atau sistem yang memperoleh keuntungan dan berada di luar dari sistem.</p>	 <p style="text-align: center;"><i>Actor</i></p>


<p><i>Use Case</i> : Mewakili sebuah bagian dari fungsionalitas sistem dan ditempatkan dalam <i>system boundary</i>.</p>	
<p><i>Subject Boundary</i> : Menyatakan lingkup dari subjek.</p>	
<p><i>Association Relationship</i>: Menghubungkan <i>actor</i> untuk berinteraksi dengan <i>use case</i>.</p>	
<p><i>Include Relationship</i> : Menunjukkan <i>inclusion</i> fungsionalitas dari sebuah <i>use case</i> dengan <i>use case</i> lainnya. arah panah dari <i>base use case</i> ke <i>included use case</i> association. Actor ditempatkan di luar <i>subject boundary</i>.</p>	<p>&lt;&lt;include&gt;&gt;</p> 
<p><i>Extend Relationship</i> : Menunjukkan <i>extension</i> dari sebuah <i>use case</i> untuk menambahkan <i>optional behavior</i>, arah panah dari <i>extension use case</i> ke <i>base use case</i>.</p>	<p>&lt;&lt;extend&gt;&gt;</p> 



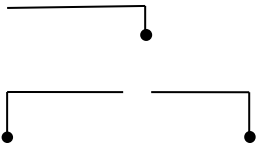
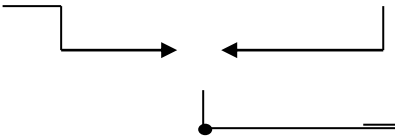
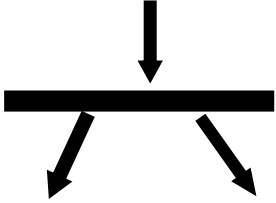
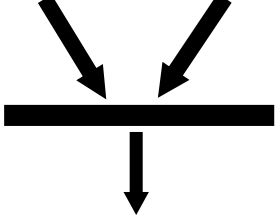
<p><i>Generalization relationship :</i></p> <p>Menunjukkan generalisasi dari <i>use case</i> khusus ke yang lebih umum.</p>	
---	--

### 2.3.2 Activity Diagram

*Activity Diagram* adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Dimana secara grafis diagram ini menggambarkan aliran *skuential* dari kegiatan baik itu proses bisnis atau sebuah *use case*. Diagram ini mempunyai peran seperti dengan *flowchart*, tetapi perbedaannya dengan *flowchart* adalah dimana diagram ini menggambarkan mekanisme kegiatan secara *paralel*. *Activity Diagram* sangat berguna untuk memodelkan kegiatan, menunjukkan tahapan, pengambilan keputusan dan percabangan yang akan dilakukan saat sebuah operasi dieksekusi dan untuk memodelkan hasil dari kegiatan tersebut (Nuryana:2017).

(Tabel 2.2 : Elemen-elemen *activity diagram*)

Notasi	Nama Elemen dan Fungsi
<p><i>Activity</i></p>	<p><i>Activity</i> : Merepresentasikan sekumpulan aktivitas.</p>
	<p><i>Control Flow</i> : Menunjukkan rangkaian dari suatu eksekusi.</p>

	<p><i>Initial Node</i> : Pertanda dari suatu awal aktivitas.</p>
	<p><i>Final Activity node</i> : Untuk menunjukkan akhir dari suatu aktivitas.</p>
	<p><i>Decision node</i> : Merepresentasikan kondisi untuk memastikan bahwa alur objek hanya bergerak dalam satu jalur.</p>
	<p><i>Merge node</i> : Digunakan untuk mengembalikan berbagai <i>decision path</i> menjadi satu.</p>
	<p><i>Fork node</i> : Digunakan untuk membagi perilaku menjadi kumpulan aktivitas yang berjalan secara paralel atau bersamaan.</p>
	<p><i>Join node</i> : Digunakan untuk menyatukan kembali kumpulan aktivitas yang berjalan secara paralel atau bersamaan.</p>

### 2.3.3 *Sequence Diagram*

*Sequence Diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh obyek dan *message* (pesan) yang diletakkan diantara *obyek-obyek* ini di dalam *use case* (Nuryana:2017).

Komponen utama pada *Sequence* diagram terdiri dari:

1. *Obyek* atau *Participant*

*Obyek* yang digambarkan dengan kotak segiempat bernama. *Obyek* diletakkan didekat bagian atas diagram dengan urutan dari kiri ke kanan. Setiap *participant* terhubung dengan garis titik-titik yang disebut *lifeline*.


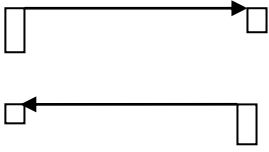
2. *Message* (Pesan)

Disimbolkan dengan sebuah garis dengan tanda panah. Sebuah *message* bergerak dari satu *participant* ke *participant* yang lain. Sebuah *participant* dapat mengirim sebuah *message* kepada dirinya sendiri.

3. *Time*

*Time* adalah diagram yang mewakili waktu pada arah *vertical*. Waktu dimulai dari atas kebawah. *Message* yang lebih dekat dari atas akan dijalankan terlebih dahulu dari pada *message* yang lebih dekat mengarah kebawah.

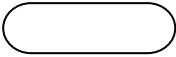



(Tabel 2.3 : Elemen-elemen *Sequence Diagram*)

No	Gambar	Nama	Keterangan
1.		<i>Life Line</i>	<i>Objek entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.

#### 2.3.4 *State Diagram*

Menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu obyek pada sistem sebagai akibat dari *stimulans* yang diterima. *State Diagram* untuk memodelkan *behavior/metode (lifecycle)* sebuah kelas atau *object* memperlihatkan urutan kejadian sesaat (*state*) yang dilalui sebuah *object*, transisi dari sebuah *state* ke *state* lainnya (Nuryana:2017).

(Tabel 2.4 : Simbol-Simbol *State Diagram*)

No	Notasi	Penjelasan
1		<i>State</i> , digambarkan berbentuk segi empat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu.
2		Titik awal ( <i>start</i> ), digunakan untuk menggambarkan awal dari kejadian dalam suatu diagram <i>statechart</i> .
3		Titik akhir ( <i>end</i> ), digunakan untuk menggambarkan akhir dari kejadian dalam suatu <i>diagram statechart</i> .
4	[ <i>Guard</i> ]	<i>Guard</i> yang meupakan syarat terjadinya transisi yang bersangkutan.
5		<i>Point</i> , digunakan untuk menggambarkan apakah akan masuk ( <i>entry point</i> ) ke dalam <i>state</i> atau akan keluar ( <i>exit point</i> )
6	<i>Event</i>	<i>Event</i> , digunakan untuk mendeskripsikan kondisi yang menyebabkan sesuatu pada <i>state</i> .



## 2.4 DataBase

Menurut Rizki (2019) *Database* merupakan kumpulan *file* yang saling berkaitan dan berinteraksi, relasi tersebut bila ditunjukkan dengan kunci dari tiap-tiap *file* yang ada. Satu database menunjukkan suatu kumpulan data yang dipakai dalam suatu lingkup perusahaan, instansi. Pengolahan database merupakan suatu cara yang dilakukan terhadap *file-file* yang berada di suatu instansi yang mana *file* tersebut dapat disusun, diurut, diambil sewaktu-waktu serta dapat ditampilkan dalam bentuk suatu laporan sehingga dapat mengelola *file-file* yang berisikan informasi tersebut secara rapi.

*Database* (basis data) adalah : Sistem penyimpanan beragam jenis data dalam sebuah entitas yang besar untuk diolah sedemikian rupa agar mudah dipergunakan kembali.

### 2.4.1 Kegunaan *DataBase*

Menurut Rizki (2019) database dibentuk untuk mengatasi persoalan yang sering dihadapi di dalam pengolahan data seperti :

#### 1. *Redudansi* dan *Inkonsistensi* Data

Penyimpanan data yang sama pada beberapa tempat atau media penyimpanan yang mengakibatkan terjadinya pemborosan media penyimpanan. Penyimpanan data yang sama dan berulang-ulang di beberapa *file* dapat mengakibatkan inkonsistensi (tidak konsisten).

#### 2. Keamanan Data

Dengan *Database* managemen, sistem keamanan data bisa dicapai. Misalnya data mengenai gaji pegawai hanya boleh dibuka oleh bagian keuangan dan personalia,

bagian lain tidak diperbolehkan menggunakannya dengan membuat suatu *password* dan wewenang atau *userauthorization* dan bersih.

### 3. Kesulitan mengakses Data

Database dapat mengakses kesulitan dalam mengakses data karena mampu mengambil data secara langsung dengan program aplikasi yang mudah digunakan.

### 4. Isolasi Data untuk Standarisasi

Jika data tersebar dalam bentuk format yang tidak sama, maka ini menyulitkan dalam menulis program aplikasi untuk mengambil dan menyimpan data. Maka suatu database haruslah dibuat suatu format.

#### **2.4.2 SQLite**

*SQLite* merupakan sebuah sistem manajemen basis data relasional yang bersifat *ACID* dan memiliki ukuran pustaka kecil, ditulis dalam bahasa c. *SQLite* merupakan proyek yang bersifat *public domain* yang dikerjakan oleh *D. Richard Hipp*. Inti *SQLite* bukanlah sebuah sistem yang mandiri yang berkomunikasi dengan program, melainkan sebagai bagian integral dari sebuah program secara keseluruhan. Seluruh elemen basis data (defenisi data, tabel, indeks dan data) disimpan sebagai sebuah *file*. Kesederhanaan dari sisi desain tersebut bisa diraih melalui cara mengunci keseluruhan *file* basis data pada saat sebuah transaksi dimulai.

#### **2.4.1 Fitur SQLite**

*SQLite* menimplementasikan hampir seluruh elemen-elemen standar yang berlaku pada SQL-92, termasuk transaksi yang bersifat atomic, konsistensi basis data, isolasi dan durabilitas (*ACID*), *trigger*, dan kueri-kueri yang kompleks.

Perancangan aplikasi sistem manajemen *SQLite* ini nantinya akan dijalankan di *smartphone* berbasis *android*. Pada awal perangkat lunak dijalankan, maka akan langsung muncul pada halaman pertama yaitu pilihan *database* yang diinginkan.

## **2.5 Sampah**

### **2.5.1 Pengertian Sampah Menurut WHO**

Menurut definisi *World Health Organization* (WHO) sampah adalah sesuatu yang tidak digunakan, tidak dipakai, tidak disenangi atau sesuatu yang dibuang yang berasal dari kegiatan manusia dan tidak terjadi dengan sendirinya.

### **2.5.2 Pengertian Sampah Menurut Undang-undang**

Undang-Undang Pengelolaan Sampah Nomor 18 tahun 2008 menyatakan sampah adalah sisa kegiatan sehari-hari manusia dan/atau dari proses alam yang berbentuk padat.

### **2.5.3 Pengertian Sampah Menurut Bahar**

Bahar (2019) berpendapat bahwa sampah adalah barang buangan berupa bahan padat yang mengakibatkan turunnya nilai estetika lingkungan, nilai sumber daya, membawa penyakit, menimbulkan polusi, menyumbat saluran air, & banyak dampak negatif lainnya.

### **2.5.4 Pengertian Sampah Menurut Kodoatie**

Kodoatie (2019) mendefinisikan sampah sebagai suatu limbah atau barang buangan yang bersifat padat ataupun setengah padat, yang berasal dari hasil dari kegiatan perkotaan atau siklus kehidupan makhluk hidup.

### 2.5.5 Pengertian Sampah Menurut Ahli Kesehatan

Para ahli kesehatan masyarakat Amerika membuat batasan, sampah (*waste*) adalah sesuatu yang tidak digunakan, tidak dipakai, tidak disenangi, atau sesuatu yang dibuang, yang berasal dari kegiatan manusia, dan tidak terjadi dengan sendirinya. Dari batasan ini jelas bahwa sampah adalah hasil kegiatan manusia yang dibuang karena sudah tidak berguna.

Dengan demikian sampah mengandung prinsip sebagai berikut :

1. Adanya sesuatu benda atau bahan padat
2. Adanya hubungan langsung/tidak langsung dengan kegiatan manusia
3. Benda atau bahan tersebut tidak dipakai lagi

## 2.6 *Kotlin*

*Kotlin* adalah sebuah bahasa pemrograman dengan pengetikan statis yang berjalan pada Mesin *Virtual Java* ataupun menggunakan kompiler LLVM yang dapat pula dikompilasikan ke dalam bentuk kode sumber *JavaScript*.

Pengembang utamanya berasal dari tim programmer *JetBrains* yang bermarkas di *Saint Petersburg*, Rusia. *JetBrains* terkenal dengan produk *IntelliJ IDEA*, sebuah *platform IDE* yang sangat *powerful* dan banyak digunakan oleh *Java Developer*. *Android Studio* merupakan sebuah IDE resmi serta khusus dibuat untuk *Android Development*. Pembuatan IDE ini berdasarkan *IntelliJ IDEA*.

*Kotlin* didesain berdasarkan pemahaman pemrograman dari *Java Developer* serta *IntelliJ* sebagai IDE utamanya. *Kotlin* sangat intuitif dan mudah dipelajari bagi *Java Developer*. Sebagian besar bahasanya sangat mirip dengan apa yang ada

pada *Java*. Para *Java Developer* pun dapat mempelajari perbedaan konsep dasar pada *Kotlin* dalam waktu singkat. Meskipun sintaksisnya tidak kompatibel dengan bahasa *Java*, *Kotlin* didesain untuk dapat bekerja sama dengan kode bahasa *Java* dan bergantung kepada kode bahasa *Java*, seperti berbagai *framework* *Java* yang ada.

Tim Pengembang memutuskan untuk memberi nama “*Kotlin*”, yakni nama sebuah pulau di Rusia, sebagaimana ”*Java*” yang berasal dari nama pulau Jawa (atau *Java* dalam bahasa Inggris) di Indonesia.

Pada *Google I/O 2017 Kotlin* resmi diumumkan sebagai bahasa pemrograman yang dapat digunakan untuk membuat aplikasi *Android*. *Google* menetapkan *Kotlin* sebagai bahasa kelas satu bagi *Android*. Maka bersama *Java* dan *C++*, *Kotlin* merupakan bahasa resmi untuk pengembangan aplikasi-aplikasi *Android*.

Dengan pengukuhan ini *Google* nyata mendukung *Kotlin* secara penuh. *Google* juga akan memastikan bahwa semua fitur baru di *Android*, *framework*, IDE dan keseluruhan *library*, akan dapat bekerja dan terintegrasi baik dengan bahasa pemrograman *Kotlin*.

Penetapan *Kotlin* menjadi bahasa *first-party* pemrograman aplikasi *Android* ini sesuai dengan aspirasi para *developer* di komunitas yang telah lama menginginkannya. Keuntungannya adalah kini kita dapat menggunakan bahasa pemrograman modern *Kotlin* untuk membuat aplikasi di *Android*.

## **2.7 Google Map API**

*Google Maps* adalah layanan gratis yang diberikan oleh *Google* dan sangat populer. *Google Maps* adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, *Google Maps* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu *browser*. Kita dapat menambahkan fitur *Google Maps* dalam *web* yang telah kita buat atau pada *blog* kita yang berbayar maupun gratis sekalipun dengan *Google Maps API*. *Google Maps API* adalah suatu *library* yang berbentuk *JavaScript* (Ahmad Syafiq, dkk : 2016).

## **2.8 Back4App**

*Back4App* adalah merupakan *Parse Server Hosting platform* untuk membangun dan mengelola aplikasi menggunakan *Parse Open Source* yang menyediakan fitur yang siap pakai untuk dapat membuat aplikasi baru atau migrasi aplikasi dari *Parse*. *Back4app* memberikan alternatif yang paling efisien dan ekonomis untuk *Parse*. *Back4app* menawarkan tingkat layanan jauh lebih unggul mengingat faktor-faktor seperti migrasi aplikasi, kinerja, kehandalan, dukungan pelanggan, perencanaan pengguna yang fleksibel dan ketersediaan banyak fitur lainnya yang menjadikan *Back4App* sebagai pilihan untuk dapat menjalankan aplikasi *Parse*. (George Batschinski, 2016)

## BAB III

### METODE PENELITIAN

#### 3.1 Tahapan Penelitian



**Gambar 3.1** *Flowchart* Tahapan Penelitian

Keterangan pada tahap perencanaan merupakan tahap awal dalam penelitian ini. Ada beberapa hal yang ditentukan dalam tahap perencanaan ini, yaitu menentukan masalah, menentukan ruang lingkup dan tujuan, dan menentukan judul. Pada tahap pengumpulan data dilakukan untuk mendapatkan suatu informasi yang dibutuhkan dalam mencapai tujuan penelitian seperti observasi, wawancara, dan dokumentasi.

Pada tahap analisa sistem dilakukan suatu analisa pada suatu sistem untuk menentukan seberapa jauh sistem tersebut telah mencapai sasarannya. Langkah-langkah yang dilakukan seperti (*Identify*) mengidentifikasi masalah, (*Understand*) memahami kerja sistem yang ada, (*Analyze*) menganalisis sistem, dan (*Report*) membuat laporan hasil analisis.

Tahap selanjutnya setelah analisa sistem, mendapatkan gambaran dengan jelas tentang apa yang dikerjakan pada analisa sistem, maka dilanjutkan dengan memikirkan bagaimana membentuk sistem tersebut. Tujuan yang hendak dicapai dari tahap perancangan sistem mempunyai maksud atau tujuan utama untuk memenuhi kebutuhan pemakaian sistem (*user*) dan untuk memberikan gambaran yang jelas dan menghasilkan rancangan bangun yang lengkap.

Sistem setelah dianalisa dan dirancang, maka sistem tersebut siap diterapkan atau diimplementasikan. Tahap implementasi adalah tahap dimana sistem informasi telah digunakan oleh pengguna. Sebelum benar-benar bisa digunakan dengan baik oleh pengguna, sistem harus melalui tahap pengujian terlebih dahulu untuk menjamin tidak ada kendala fatal yang muncul pada saat pengguna memanfaatkan sistemnya.

Selanjutnya tahap pengujian sistem dilakukan Proses pengujian berfokus pada logika internal software atau aplikasi, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan – kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan. Dan setelah semua tahapan dilakukan, tahap hasil atau laporan penelitian ini



merupakan tahap akhir dalam sebuah proses penelitian. Untuk kepentingan publikasi dibuat hasil berupa Bentuk dan sistematik laporan penelitian dapat berupa laporan atau skripsi.

### **3.2 Metode Pengumpulan Data**

Metode pengumpulan data berupa suatu pernyataan tentang sifat, keadaan, kegiatan tertentu dan sejenisnya. Pengumpulan data dilakukan untuk mendapatkan suatu informasi yang dibutuhkan dalam mencapai tujuan penelitian. Dalam penyusunan laporan penelitian ini penulis mengambil objek penelitian di Kota Medan.

Pengumpulan data dalam penelitian pada penelitian ini menggunakan 3 cara berikut merupakan uraian yang digunakan :

1. Observasi

Suatu metode pengumpulan data yang dilakukan dengan mengamati langsung, melihat dan mengambil suatu data yang dibutuhkan di tempat Penelitian itu dilakukan. Observasi juga bisa diartikan sebagai proses yang kompleks. Pengumpulan data yang dilakukan di Kota Medan.

2. Wawancara

Wawancara merupakan salah satu teknik pengumpulan data yang dilakukan melalui tatap muka langsung dengan narasumber dengan cara tanya jawab langsung. Wawancara dilakukan dengan petugas kebersihan dan masyarakat sekitar Kota Medan yang berhubungan dengan data yang terkait.

### 3. Dokumentasi

Suatu pengumpulan data dengan cara melihat langsung sumber-sumber Dokumen yang terkait. Dengan arti lain bahwa dokumentasi sebagai Pengambilan data melalui dokumen tertulis maupun elektronik, digunakan sebagai mendukung kelengkapan data yang lain.

### **3.3 Analisis Sistem Sedang Berjalan**

Dalam proses pembuatan suatu sistem harus dilakukan penelitian dan penganalisaan tentang sistem sistem yang berjalan dan yang akan dibangun. Analisis sistem merupakan gambaran tentang sistem yang saat ini sedang berjalan di dinas kebersihan Kota Medan pada bagian pengangkutan sampah , sistem yang digunakan masih sederhana dan manual yaitu dengan menggunakan media telepon sebagai alat untuk menelpon/melaporkan ke dinas kebersihan bahwasanya sampah belum diangkut. Sehingga diusulkan analisis sistem yang bertujuan untuk membuat sistem yang baru agar terkomputerisasi sehingga dapat lebih efektif dan efisien.

#### 3.3.1 Kelemahan Sistem Yang Berjalan

Sistem yang digunakan masih sederhana dan manual yaitu dengan menggunakan media telepon sebagai alat pelapor untuk menelpon/melaporkan ke dinas kebersihan bahwasanya sampah belum diangkut. Dalam proses pelaporannya, pihak dinas kebersihan masih direpotkan untuk membuat data pelapor dan laporan tempat atau lokasi sampah yang di laporkan.

### 3.3.2 Gambaran Umum Sistem yang Diusulkan

Gambaran umum dari sistem yang diusulkan adalah untuk memberikan gambaran secara umum kepada pengguna (*user*) mengenai sistem aplikasi berbasis *android*, perancangan sistem secara umum juga sudah dapat mengenai komponen sistem aplikasi yang akan didesain. Penentuan persyaratan sistem dilakukan agar arah perancangan sistem dapat terarah pada sasaran, oleh sebab itu sistem yang dirancang harus memenuhi batasan sistem dimana perancangan sistem aplikasi ini merupakan kebutuhan fungsional.

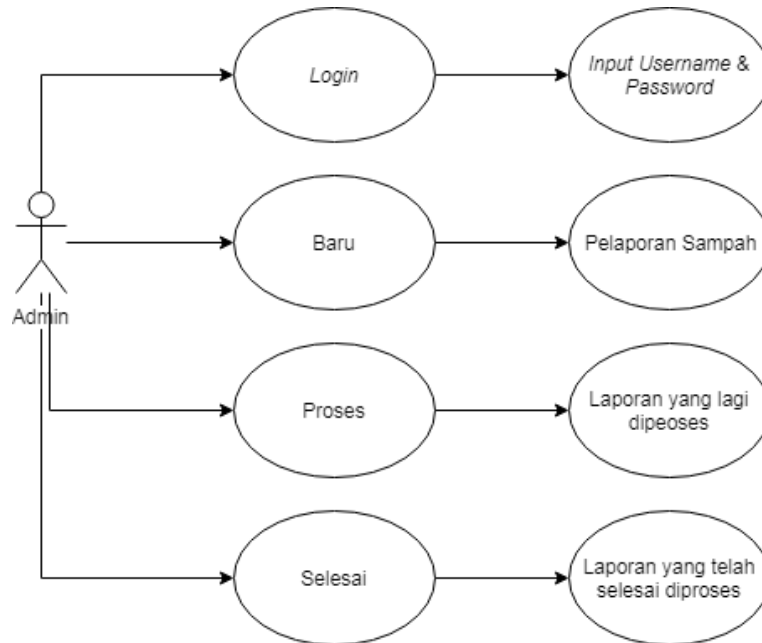
Implementasi menggambarkan bagaimana suatu sistem di bentuk. Pada tahap perancangan penelitian aplikasi android ini dirancang dengan tujuan membuat tampilan aplikasi agar mudah dimengerti pengguna dalam pengaduan masyarakat untuk pembuangan sampah yang menumpuk dan dapat mengolah data rekapitulasi sampah yang telah diangkut atau belum diangkut

## 3.4 Rancangan Penelitian

### 3.4.1 *Use Case Diagram*

*Use case diagram* menggambarkan aktifitas yang dilakukan oleh suatu sistem, yang dimana proses transaksi melalui *android* yang dilakukan oleh *admin*, antara lain:

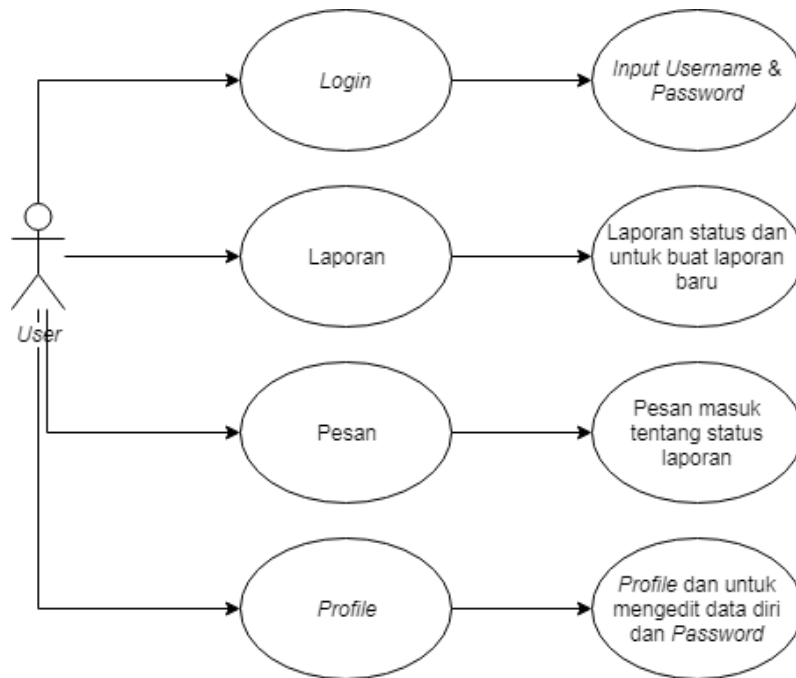
### 3.4.1.1 User Diagram



**Gambar 3.2** Use Case Diagram Admin

keterangan Use case diagram admin :

- a. *Login* : Merupakan menu untuk menginput *Username* dan *Password* akun *admin*.
- b. *Baru* : Merupakan menu yang menampilkan laporan-laporan baru dari para *user*.
- c. *Proses* : Merupakan menu yang menampilkan laporan-laporan *user* yang lagi di proses.
- d. *Selesai* : Merupakan menu yang menampilkan laporan-laporan *user* yang telah selesai di proses.



**Gambar 3.3** *Use Case Diagram User*

keterangan *Use case diagram user* :

- a. Login : Menu untuk menginput *Username* dan *Password* dari akun *user*.
- b. Laporan : Menu untuk menampilkan tentang status laporan yang telah dikirim dan untuk membuat laporan baru.
- c. Pesan : Menu untuk pesan masuk dari petugas tentang status laporan yang telah dikirim dan yang telah selesai.
- d. Profile : Menu untuk melihat *profile* dan mengedit data diri dan *password*.

### 3.4.2 Activity Diagram

*Activity diagram* digunakan untuk memberikan gambaran mengenai alur aktivitas di dalam sistem.

### 3.4.2.1 Activity Diagram Admin

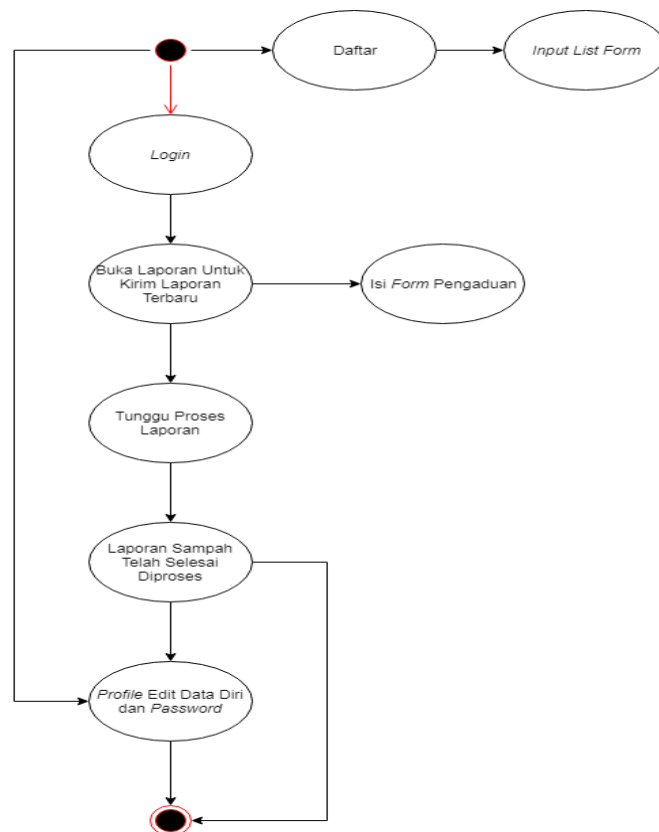


**Gambar 3.4** Activity Diagram Admin

Keterangan *Activity diagram admin* :

1. *Admin* terlebih dahulu melakukan login agar bisa masuk ke sistem aplikasi.
2. Kemudian *admin* memproses laporan yang masuk.
3. Setelah itu *admin* mengganti status laporan yang lagi di proses tadi menjadi selesai jika pengerjaan laporan tadi telah selesai dikerjakan.
4. Lalu *admin* bisa melihat semua laporan yang telah selesai di kerjakan di menu laporan.

### 3.4.2.2 *Activity Diagram User*

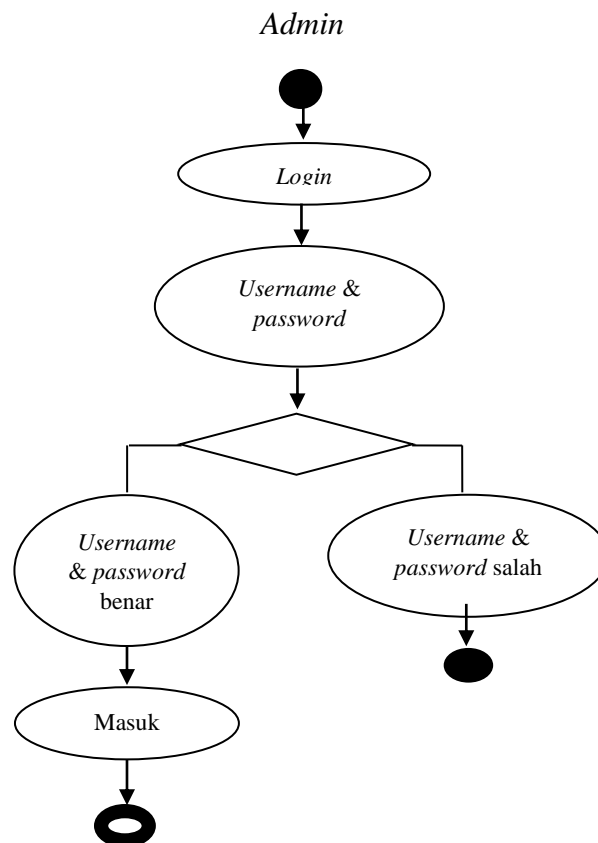


**Gambar 3.5** *Activity Diagram User*

Keterangan *Activity diagram user* :

1. *User* melakukan pendaftaran terlebih dahulu untuk membuat akun
2. Kemudian setelah melakukan pendaftaran, *user login* untuk bisa masuk ke dalam sistem aplikasi.
3. Buka menu laporan dan tekan menu kirim laporan baru untuk membuat laporan sampah.
4. Isi *form* yang ada di menu kirim laporan baru tersebut.
5. Setelah itu tunggu hingga petugas memproses laporan dan melakukan tugas mereka sampai selesai.
6. Setelah selesai, laporan yang tadi dikirim otomatis berstatus selesai.
7. Di menu profil anda bisa melihat profil dan mengedit data diri dan *password*.

### 3.4.2.3 *Activity Diagram Login Admin*



**Gambar 3.6** *Activity Diagram Login Admin*

Keterangan *Activity diagram login admin* :

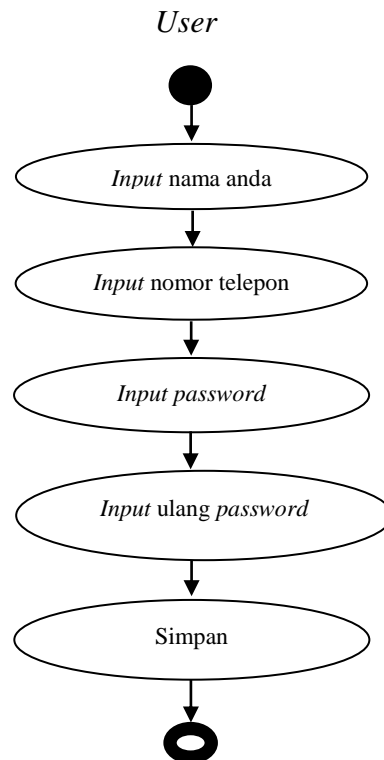
1. *Admin* ke menu *login* terlebih dahulu



2. Setelah itu *admin* menginput *username & password*
3. Lalu akan diproses, jika benar maka *admin* akan masuk ke sistem aplikasi.

Jika salah, maka akan ada pemberitahuan *username & password* salah.

#### 3.4.2.4 Activity Diagram Daftar User

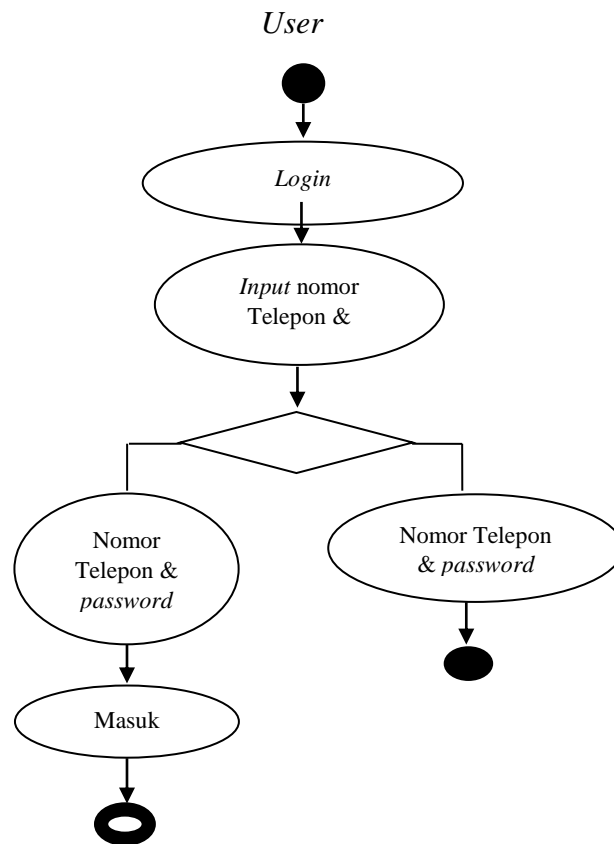


**Gambar 3.7** Activity Diagram Daftar User

Keterangan Activity diagram login admin :

1. Masuk ke menu daftar
2. *Input* nama anda
3. *Inputkan* nomor telepon anda
4. *Inputkan password* yang ingin anda gunakan untuk sistem aplikasinya
5. *Input ulang password* yang anda buat sebelumnya

### 3.4.2.5 Activity Diagram Login User



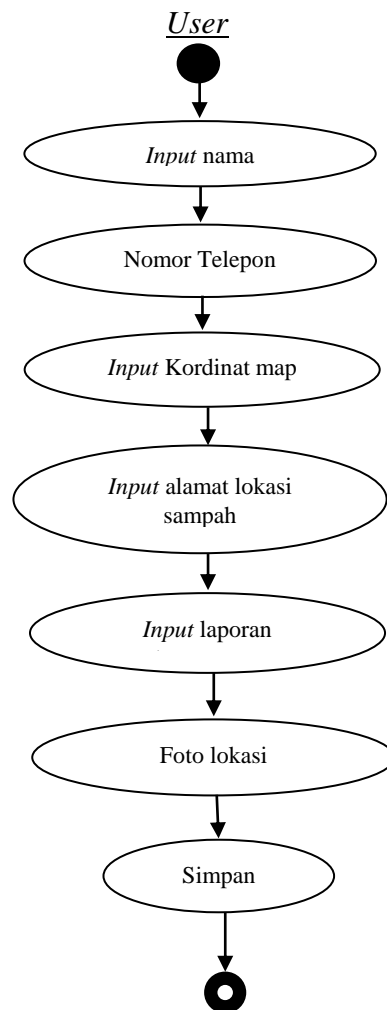
**Gambar 3.8** Activity Diagram Login

Keterangan Activity diagram login user :

1. *User* ke menu *login* terlebih dahulu
2. Setelah itu *user* menginput nomor telepon & *password*
3. Lalu akan diproses, jika benar maka *user* akan masuk ke sistem aplikasi.

Jika salah, maka akan ada pemberitahuan nomor telepon & *password* salah.

### 3.4.2.6 Activity Diagram Laporan Baru



**Gambar 3.9** Activity Diagram Laporan Baru

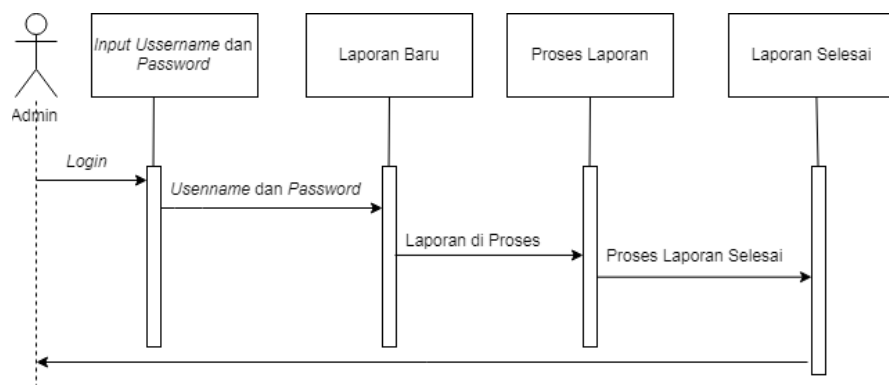
Keterangan Activity diagram laporan baru :

1. *Input* nama anda. Tapi biasanya *form* ini otomatis terisi sendiri dan di sesuaikan dengan nama *user*.
2. *Input* nomor telepon. Dan ini juga otomatis terisi sendiri sesuai nomor telepon *user* yang sudah di daftarkan.
3. *Input* kordinat *map*. Cari kordinat sesuai alamat yang akan dilaporkan.
4. *Input* alamat lokasi.

5. *Input* laporan keterangan pengaduan.
6. *Input* foto lokasi tempat.
7. Kemudian simpan data tersebut.

### 3.5 *Sequence Diagram*

#### 3.5.1 *Sequence Diagram Admin*

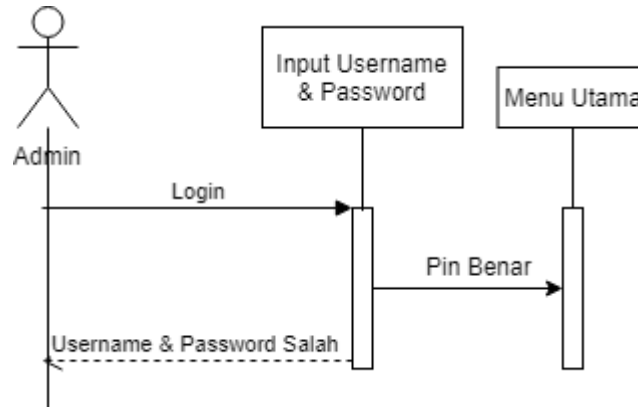


**Gambar 3.10** *Sequence Diagram Admin*

Keterangan Sequence Diagram *Admin* :

1. *Admin* melakukan login terlebih dahulu ke sistem aplikasi.
2. *Admin* melihat daftar laporan pengaduan sampah yang masuk.
3. Kemudian *admin* memproses laporan tersebut agar segera di kerjakan oleh petugas.
4. Setelah petugas melakukan pekerjaannya, *admin* harus mengajukan bahwa laporan yang di proses tadi telah selesai.

### 3.5.2 Sequence Diagram Login Admin

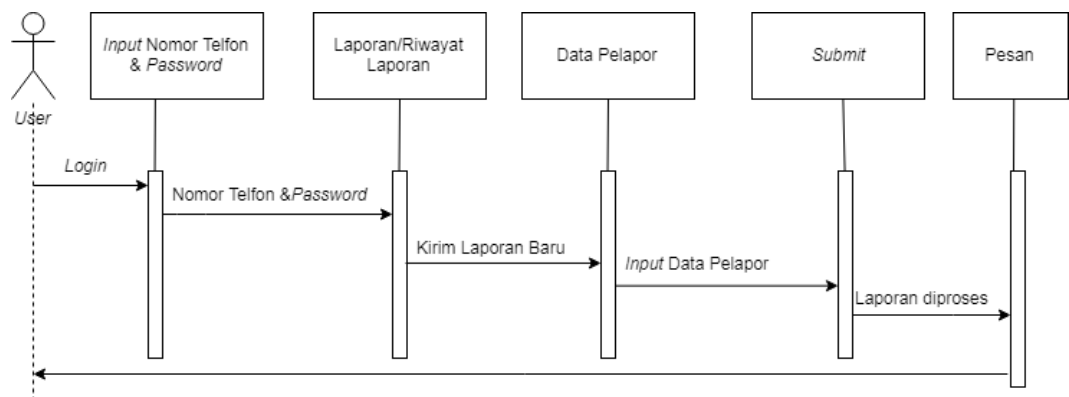


**Gambar 3.11** Sequence Diagram Login Admin

Keterangan Sequence Diagram Login Admin :

1. Admin login ke sistem aplikasi dengan menginput *username* dan *password*.
2. Jika *username* dan *password* benar, admin akan masuk ke menu utama sistem pada aplikasi, tetapi jika *username* dan *password* salah, maka admin akan kembali ke menu login.

### 3.5.3 Sequence Diagram User

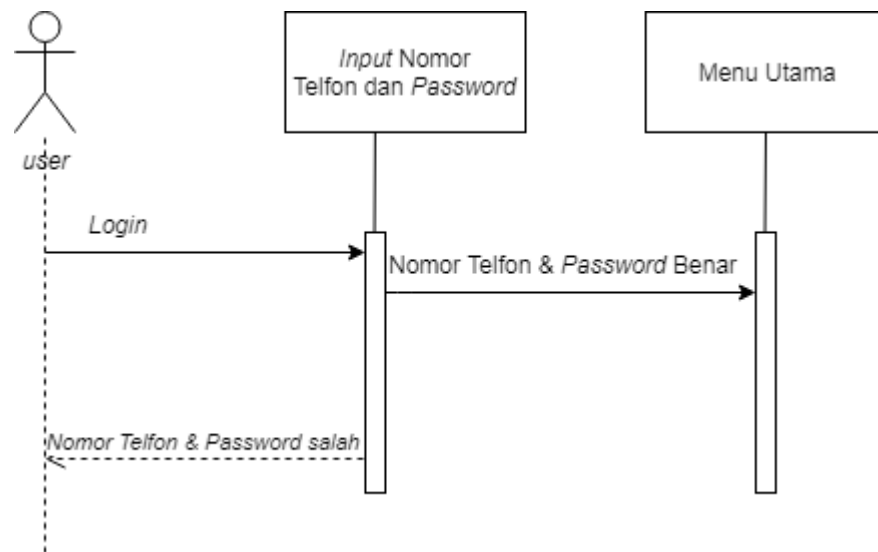


**Gambar 3.12** Sequence Diagram User

Keterangan *Sequence Diagram User* :

1. *User* melakukan *login* terlebih dahulu ke sistem aplikasi
2. Setelah *user* masuk, *user* akan ke menu utama yaitu menu laporan/riwayat laporan.
3. Kemudian untuk membuat laporan terbaru, *user* harus menuju atau memilih menu kirim laporan baru untuk membuat laporan terbaru.
4. Setelah itu *user* menginput data pelapor, kemudian tekan *submit*.
5. Dan kemudian *user* tinggal menunggu laporan di proses.

#### 3.5.4 *Sequence Diagram Login User*



**Gambar 3.13** *Sequence Diagram Login User*

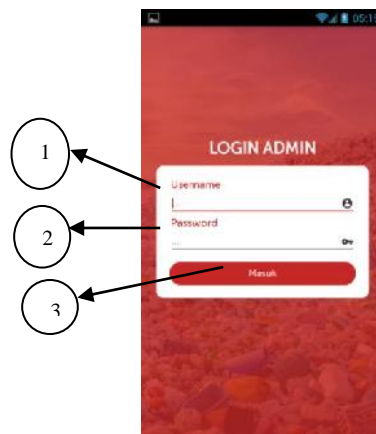
Keterangan *Sequence Diagram Login User* :

1. *User* login ke sistem aplikasi dengan menginput nomor telepon dan *password*.

2. Jika nomor telepon dan *password* benar, *user* akan masuk ke menu utama sistem pada aplikasi, tetapi jika nomor telepon dan *password* salah, maka *user* akan kembali ke menu login.

### 3.6 Perancangan Aplikasi *Android*

#### 3.6.1 Tampilan Awal Aplikasi *Admin*

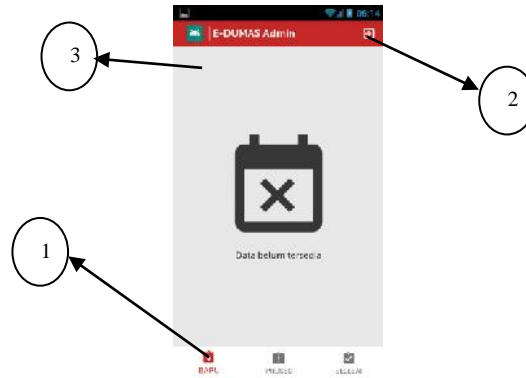


**Gambar 3.14** Tampilan Awal Aplikasi Admin

Keterangan Tampilan Aplikasi :

1. *Username* : adalah nama pengguna yang wajib diisi untuk masuk ke sistem aplikasi.
2. *Password* : Kata sandi untuk masuk ke dalam sistem aplikasi.
3. Masuk : Menu untuk masuk ke dalam sistem aplikasi apabila jika *username* dan *passwordnya* benar.

### 3.6.2 Tampilan Menu Baru di Aplikasi *Admin*

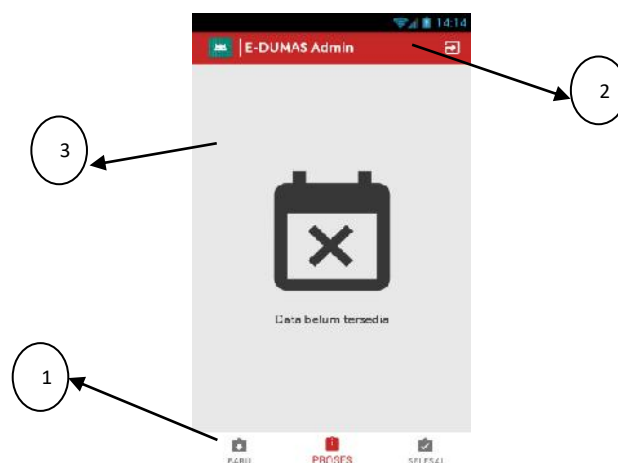


**Gambar 3.15** Tampilan Awal Aplikasi *Admin*

Keterangan Tampilan Aplikasi :

1. Baru : adalah Menu halaman laporan yang masuk.
2. *Log out* : adalah proses keluar dari sistem aplikasi
3. Halaman tampilan : Halaman tampilan untuk menampilkan laporan pengajuan/pengaduan.

### 3.6.3 Tampilan Menu Proses di Aplikasi *Admin*



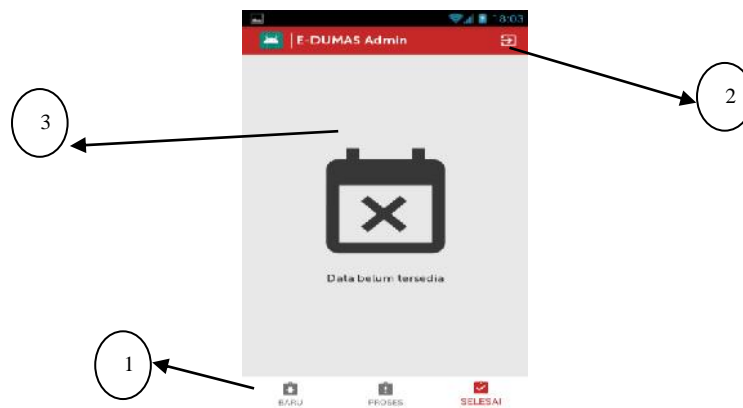
**Gambar 3.16** Tampilan Menu Proses



Keterangan Tampilan Aplikasi :

1. Proses : adalah Menu halaman laporan yang lagi di proses.
2. *Log out* : adalah proses keluar dari sistem aplikasi
3. Halaman tampilan : Halaman tampilan untuk menampilkan laporan yang sedang di proses.

### 3.6.4 Tampilan Menu Selesai di Aplikasi *Admin*

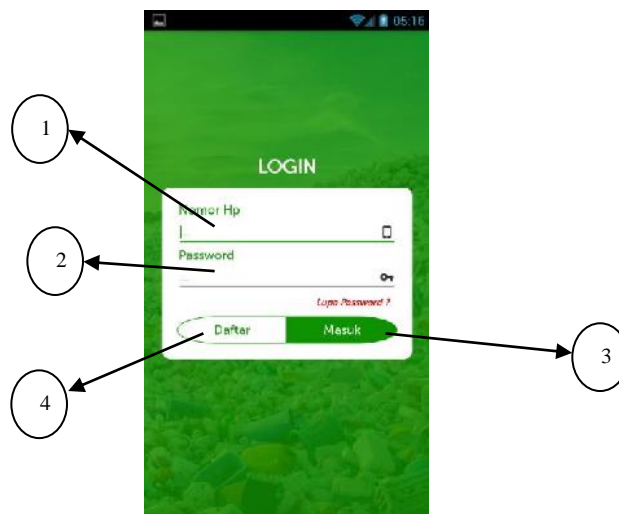


**Gambar 3.17** Tampilan Menu Selesai

Keterangan Tampilan Aplikasi :

1. Selesai : adalah Menu halaman laporan yang telah selesai.
2. *Log out* : adalah proses keluar dari sistem aplikasi
3. Halaman tampilan : Halaman tampilan untuk menampilkan laporan yang telah selesai.

### 3.6.5 Tampilan Menu Awal Aplikasi *User*

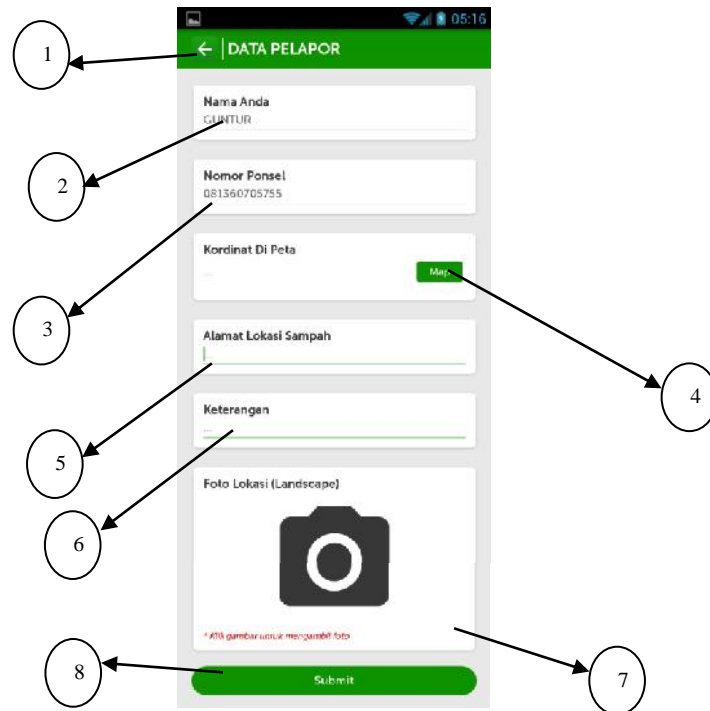


**Gambar 3.18** Tampilan Awal Aplikasi *User*

Keterangan Tampilan Aplikasi :

1. Nomor Hp : Nomor telepon Pengguna/*User* yang telah di daftarkan.
2. *Password* : Kata sandi untuk masuk ke dalam sistem aplikasi.
3. Masuk : Menu untuk masuk ke dalam sistem aplikasi apabila jika *username* dan *passwordnya* benar.
4. Daftar : adalah Menu untuk menuju *form* pendaftaran *user*.

### 3.6.6 Tampilan *form* di Menu Daftar.

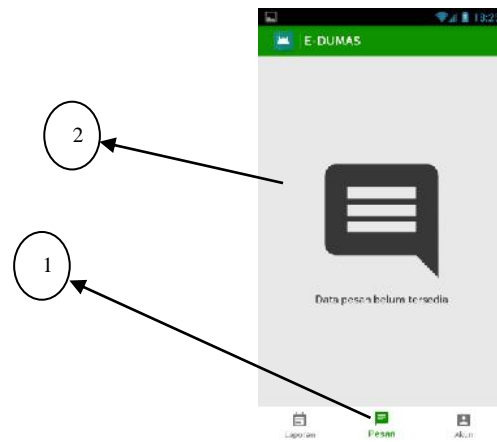


**Gambar 3.19** Tampilan *form* di Menu Daftar

Keterangan Tampilan Aplikasi :

1. *Back* : Untuk kembali ke menu sebelumnya
2. Nama anda: adalah nama *user* atau nama si pelapor.
3. Nomor Ponsel : Nomor ponsel si pelapor/*user*.
4. *Map* : Menu untuk menentukan kordinat di pelapor/*user*.
5. Alamat Lengkap : Alamat si pelapor/*user*.
6. Keterangan : Keterangan laporan si pelapor/*user*
7. Foto Lokasi : Foto lokasi pelaporan
8. Submit : Untuk menyimpan data yang telah diisi si pelapor/*user*.

### 3.6.7 Tampilan Menu Pesan.

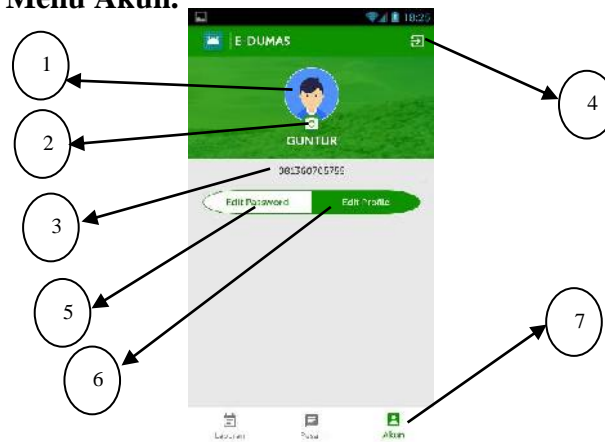


**Gambar 3.20** Tampilan Menu Pesan

Keterangan Tampilan Aplikasi :

1. Pesan : Menu halaman laporan baik yang sedang di proses maupun yang telah selesai.
2. Halaman tampilan : Halaman tampilan untuk menampilkan baik laporan yang sedang di proses maupun yang telah selesai.

### 3.6.8 Tampilan Menu Akun.

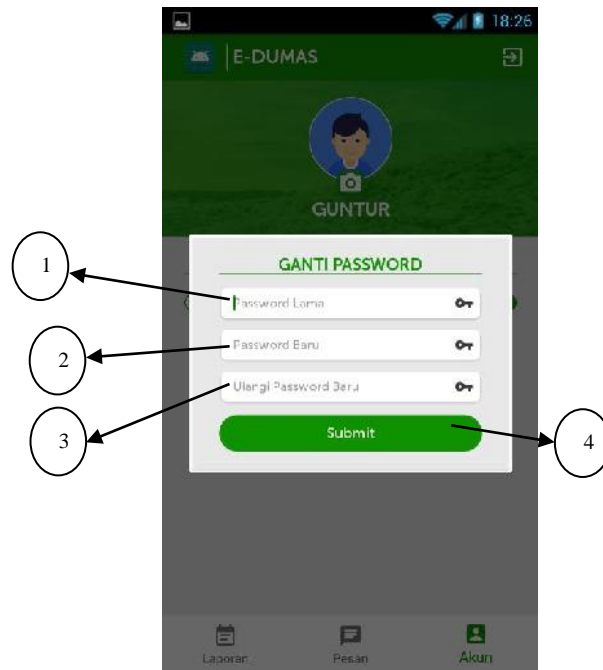


**Gambar 3.21** Tampilan Menu Akun

Keterangan Tampilan Aplikasi :

1. Foto Profil : Untuk menampilkan foto profil si *user*.
2. Gambar Kamera : Untuk mengganti foto profil si *user*.
3. Nomor Telepon : Menampilkan nomor ponsel si *user*.
4. *Log out* : adalah proses keluar dari sistem aplikasi.
5. *Edit Password* : Untuk ke menu mengubah *password* si *user*.
6. *Edit Profile* : Untuk mengedit nama atau mengganti nama *user*.
7. Halaman tampilan : Halaman tampilan untuk menampilkan baik laporan yang sedang di proses maupun yang telah selesai.

### 3.6.9 Tampilan Ganti Password.

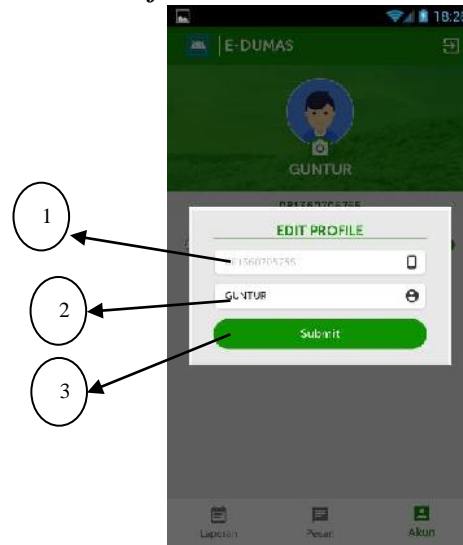


**Gambar 3.22** Tampilan Ganti Password

Keterangan Tampilan Aplikasi :

1. Password Lama : Untuk menampilkan foto profil si *user*.
2. Password Baru : Untuk mengganti foto profil si *user*.
3. Ulangi Password Baru : Menampilkan nomor ponsel si *user*.
4. Submit : Untuk menyimpan data yang telah diedit si *user*.

### 3.6.10 Tampilan Edit Profile

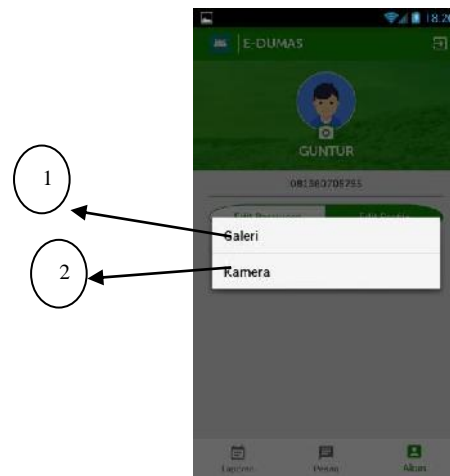


**Gambar 3.23** Edit Profile

Keterangan Tampilan Aplikasi :

1. Nomor Telepon : Menampilkan nomor telepon si *user*.
2. Nama : Mengganti nama si *user*.
3. *Submit* : Untuk menyimpan data yang telah edit si *user*.

### 3.6.11 Tampilan Edit Foto Profile



**Gambar 3.24** Edit Foto Profile

Keterangan Tampilan Aplikasi :

1. Galeri : Untuk mengganti foto profil melalui galeri.
2. Kamera : Untuk mengganti foto profil melalui media kamera



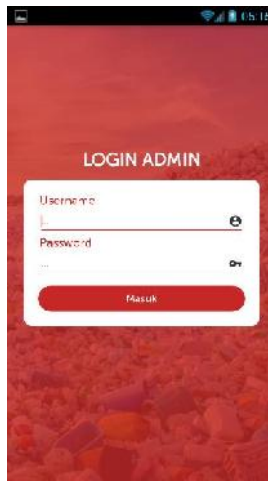
## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Hasil Tampilan Pada Sistem *Admin*

##### 4.1.1 Tampilan Halaman Utama di Sistem *Admin*

Berikut ini adalah tampilan *form login admin* untuk masuk ke sistem yang ada di *admin*.



Gambar 4.1 Tampilan *Login Admin*

##### 4.1.2 Tampilan Halaman Laporan Baru

Setelah *admin login* dan berhasil, maka akan tampil tampilan berikut ini.

Ini adalah tampilan untuk menampilkan laporan-laporan baru.



**Gambar 4.2 Halaman Laporan Baru**

#### **4.1.3 Tampilan Halaman Proses**

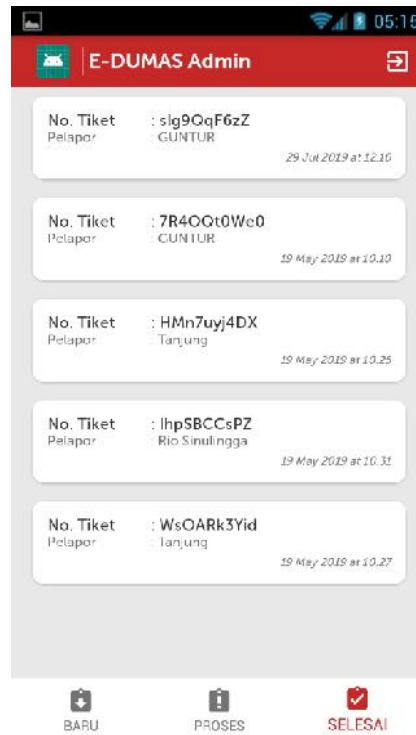
Setelah itu ada menu proses. Berikut ini adalah gambar menu proses.



**Gambar 4.3 Halaman Laporan Baru**

#### **4.1.4 Tampilan Halaman Proses**

Berikut ini adalah tampilan halaman selesai pada *admin*. Tampilan ini menunjukkan laporan-laporan yang telah selesai di kerjakan.

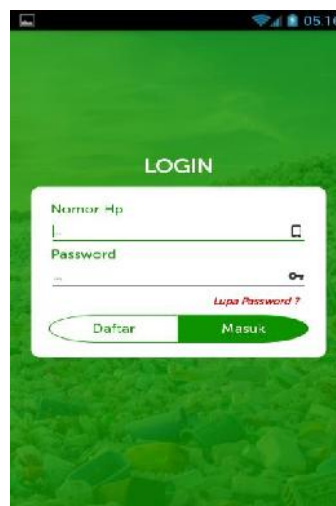


**Gambar 4.4 Halaman Selesai**

## 4.2 Hasil Tampilan Pada Sistem *User*

### 4.2.1 Tampilan Halaman Utama di Sistem *Admin*

Berikut ini adalah tampilan *form login user* untuk masuk ke sistem yang ada di *user*.



**Gambar 4.5 Halaman *login User***

#### 4.2.2 Tampilan Halaman Utama Daftar *User*

Sebelum *login*, *user* terlebih dahulu mendaftarkan diri agar bisa mengakses sistem yang ada di sistem *user*. Berikut adalah tampilan *form* pendaftaran dari *user*.



**Gambar 4.6 Halaman Utama Daftar *User***

#### 4.2.3 Tampilan Halaman Laporan/Riwayat Laporan

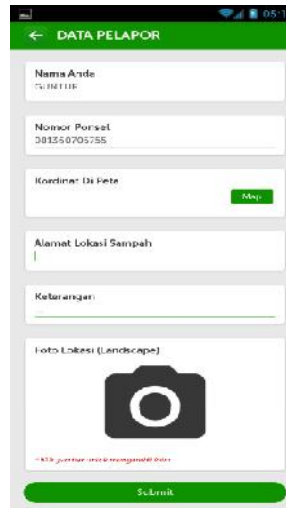
Selanjutnya adalah tampilan ketika *user* berhasil masuk ke sistem, ini adalah tampilan halaman laporan ataupun riwayat laporan. Disini *user* bisa melihat laporan-laporan yang telah selesai maupun yang lagi diproses. Dan dari sinilah *user* bisa mengirim laporan untuk pengaduan sampah.



**Gambar 4.7 Halaman Laporan/Riwayat Laporan**

#### 4.2.4 Tampilan Halaman Kirim Laporan Baru

Berikutnya adalah tampilan halaman dari menu kirim laporan baru. Disini *user* akan mengisi *form* data untuk laporan pengaduan sampah.



The screenshot shows a mobile application interface for reporting waste. The title bar is green with a white arrow and the text "DATA PELAPOR". Below the title bar, there are several input fields: "Nama Anda" with the value "75.081118", "Nomor Ponsel" with the value "081350705755", "Koordinat Di Peta" with a green "Map" button to its right, "Alamat Lokasi Sampah", "Keterangan", and "Foto Lokasi (Landscape)" which features a camera icon. At the bottom of the form is a green button labeled "Simpan".

**Gambar 4.8 Halaman Kirim Laporan Baru**

#### 4.2.5 Tampilan Halaman Pesan

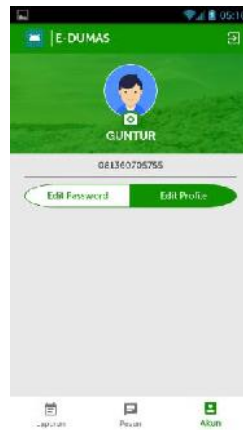
Disini kita bisa melihat pesan-pesan masuk tentang status laporan-laporan yang telah kita ajukan tadi dari *admin*/petugas.



**Gambar 4.9 Halaman Pesan**

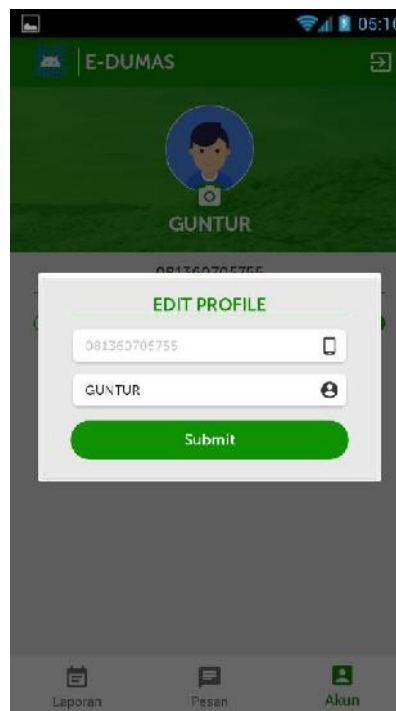
#### 4.2.6 Tampilan Akun User

Ini adalah tampilan halaman dari akun *user*. Disini *user* bisa mengedit data diri, termasuk bisa ganti nama, *password* dan termasuk foto *profile*.



**Gambar 4.10 Halaman Akun User**

Dibawah ini adalah tampilan untuk mengedit data diri ataupun nama *user*.



**Gambar 4.11 Tampilan Edit Profile**

Dan ini adalah tampilan untuk merubah atau mengganti *password user*. Pada tampilan ini *user* dapat mengubah dan mengganti *password* sesuai dengan keinginan *user*.



**Gambar 4.12 Tampilan Merubah password**

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Setelah penulis menyelesaikan penelitian ini, maka dapat didapatkan kesimpulan sebagai berikut :

1. Dengan memanfaatkan aplikasi berbasis *android* ini, penumpukan sampah lebih terbantu terutama dalam hal kebersihan.
2. Dengan memanfaatkan aplikasi berbasis *android* ini, admin lebih mudah melakukan rekapitulasi data laporan pengaduan yang masuk
3. Mengaplikasikan ataupun menerapkan aplikasi ini tidaklah sulit bagi *user* dan juga mudah dipahami bagi penggunanya.
4. Berdasarkan pengujian diperoleh, menu-menu dalam aplikasi layanan tersebut berjalan dengan baik dan telah memenuhi keinginan pengguna

#### 5.2 Saran

- a. Untuk pengembangan selanjutnya diharapkan aplikasi ini bisa diimplementasikan secara luas dan bisa lebih maju lagi.
- b. Aplikasi ini dapat ditambahkan fitur *chat* atau *call* secara *online* yang dapat menambah kemampuan aplikasi dalam *monitoring* laporan pengaduan sampah.
- c. Aplikasi ini dapat ditambahkan fitur-fitur *tracking* untuk mengetahui pada saat pihak kebersihan mendatangi lokasi sampah yang akan diangkut.



## DAFTAR PUSTAKA

- Andrian, Yudhi, and Purwa Hasan Putra. "Analisis Penambahan Momentum Pada Proses Prediksi Curah Hujan Kota Medan Menggunakan Metode Backpropagation Neural Network." *Seminar Nasional Informatika (SNIf)*. Vol. 1. No. 1. 2017.
- Azmi, Fadhillah, and Winda Erika. "Analisis keamanan data pada block cipher algoritma Kriptografi RSA." *CESS (Journal of Computer Engineering, System and Science)* 2.1: 27-29.
- Barus, S., Sitorus, V. M., Napitupulu, D., Mesran, M., & Supiyandi, S. (2018). Sistem Pendukung Keputusan Pengangkatan Guru Tetap Menerapkan Metode Weight Aggregated Sum Product Assesment (WASPAS). *Jurnal Media Informatika Budidarma*, 2(2).
- Batubara, Supina. "Analisis perbandingan metode fuzzy mamdani dan fuzzy sugeno untuk penentuan kualitas cor beton instan." *IT Journal Research and Development* 2.1 (2017): 1-11.
- Dhany, H. W., Izhari, F., Fahmi, H., Tulus, M., & Sutarman, M. (2017, October). Encryption and decryption using password based encryption, MD5, and DES. In *International Conference on Public Policy, Social Computing and Development 2017 (ICOPOSDev 2017)* (pp. 278-283). Atlantis Press.
- Erika, Winda, Heni Rachmawati, and Ibnu Surya. "Enkripsi Teks Surat Elektronik (E-Mail) Berbasis Algoritma Rivest Shamir Adleman (RSA)." *Jurnal Aksara Komputer Terapan* 1.2 (2012).
- Fachri, B. (2018). Perancangan Sistem Informasi Iklan Produk Halal Mui Berbasis Mobile Web Menggunakan Multimedia Interaktif. *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, 3, 98-102.
- Fuad, R. N., & Winata, H. N. (2017). Aplikasi keamanan file audio wav (waveform) dengan terapan algoritma RSA. *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, 1(2), 113-119.
- Ginting, G., Fadlina, M., Siahaan, A. P. U., & Rahim, R. (2017). Technical approach of TOPSIS in decision making. *Int. J. Recent Trends Eng. Res*, 3(8), 58-64.
- Hafni, Layla, and Rismawati Rismawati. "Analisis faktor-faktor internal yang mempengaruhi nilai perusahaan pada perusahaan manufaktur yang terdaftar di BEI 2011-2015." *Bilancia: Jurnal Ilmiah Akuntansi* 1.3 (2017): 371-382.
- Hamdi, Muhammad Nurul, Evi Nurjanah, and Latifah Safitri Handayani. "Community development based on Ibnu Khaldun thought, sebuah interpretasi program pemberdayaan UMKM di bank zakat el-zawa." *EL MUHASABA: Jurnal Akuntansi (e-journal)* 5.2 (2014): 158-180.

- Hariyanto, E., Lubis, S. A., & Sitorus, Z. (2017). Perancangan prototipe helm pengukur kualitas udara. *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, 1(1).
- Hartanto, S. (2017). Implementasi fuzzy rule based system untuk klasifikasi buah mangga. *TECHSI-Jurnal Teknik Informatika*, 9(2), 103-122.
- Harumy, T. H. F., & Sulistianingsih, I. (2016). Sistem penunjang keputusan penentuan jabatan manager menggunakan metode mfep pada cv. Sapo durin. In *Seminar Nasional Teknologi Informasi dan Multimedia* (pp. 6-7).
- Havena, M., & Marlina, L. (2018). The Technology of Corn Processing as an Effort to Increase The Income of Kelambir V Village. *Journal of Saintech Transfer*, 1(1), 27-32.
- Kusniyati, H., Sitanggang, N.S.P. (2016). Aplikasi Edukasi Budaya Toba Samosir Berbasis *Android*. *Jurnal Teknik Informatika*. Vol. 9 No. 1. ISSN : 1979-9160.
- Maiyana, E. (2018). Pemanfaatan *Android* Dalam Perancangan Aplikasi Kumpulan Doa. *Jurnal Sains dan Informatika*. E-ISSN : 2502-096X. P-ISSN : 2459-9549.
- Nuryana, Y., Mulyani, A. (2017). Pengembangan Aplikasi Pengendalian Skripsi Berbasis *Android* Untuk Mahasiswa dan Dosen. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*. Vol. 14 No. 2. ISSN : 2302-7339.
- Riyono, J. Sistem Informasi Pemesanan Lapangan Futsal Berbasis Aplikasi *Android* Dengan Metode Transaksi Menggunakan Kartu RFID. *Skripsi*. Universitas Teknologi Yogyakarta.
- Rahmanto, H.H.D. (2017). Alat Pengontrol dan Pemantau Lampu Penerangan Dengan Menggunakan *Android*. *Skripsi*. Universitas Katolik Widya Mandala Surabaya.
- Sulistiyorini, N.R., Darwis. R.S., Gutama, A.S. Partisipasi Masyarakat Dalam Pengelolaan Sampah di Lingkungan Margaluyu Kelurahan Cicurug. *Share Social Work Jurnal*. Vol. 5 No. 1. ISSN : 2339-0042.
- Rizki. (2019). Perancangan Aplikasi Transaksi Penjualan Di Grosir Rolli Rambung Binjai Berbasis *Android*. *Skripsi*. Universitas Pembangunan Panca Budi.
- Rizal, M. Analisis Pengelolaan Persampahan Perkotaan. *Jurnal SMARTek*. Vol. 9 No.2 Mei 2011: 155-172.
- Sifauttijani, F., Listyorini, T., Meimaharani, R. (2017). Pencarian Rumah Makan Berbasis *Android*. *Jurnal SIMETRIS*. Vol. 8 No. 1. ISSN : 2252-4983.