

# **PENGEMBANGAN APLIKASI TEMPAT SAMPAH CERDAS BERBASIS MOBILE**



Disusun Oleh:

N a m a : Adelia Nur Pratiwi  
NIM : 22523089

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

**2026**

**HALAMAN PENGESAHAN DOSEN PEMBIMBING**

**PENGEMBANGAN APLIKASI TEMPAT SAMPAH  
CERDAS BERBASIS MOBILE**

**TUGAS AKHIR**



Yogyakarta, 24 April 2026

Pembimbing,

( IR. Kurniawan Dwi Irianto, S.T., M.SC. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

# PENGEMBANGAN APLIKASI TEMPAT SAMPAH CERDAS BERBASIS MOBILE

## TUGAS AKHIR

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

Yogyakarta, 3 Juni 2026

Tim Penguji

IR. Kurniawan Dwi Irianto, S.T., M.SC.

**Anggota 1**

Taufiq Hidayat, S.T., M.C.S., Ph.D.

**Anggota 2**

Kholid Haryono, S.T., M.Kom.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



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

## 1. HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Adelia Nur Pratiwi  
NIM : 22523089

Tugas akhir dengan judul:

### **PENGEMBANGAN APLIKASI TEMPAT SAMPAH CERDAS BERBASIS MOBILE**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian 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 risiko dan konsekuensi apapun.

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

Yogyakarta, 24 April 2026

  
( Adelia Nur Pratiwi )

## HALAMAN PERSEMBAHAN

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “Pengembangan Aplikasi Tempat Sampah Cerdas Berbasis *Mobile*”. Karya ini penulis persembahkan sebagai bentuk rasa terima kasih yang sebesar-besarnya kepada keluarga besar tercinta, khususnya kedua orang tua yang senantiasa memberikan doa, dukungan, perhatian, serta pengorbanan yang tiada henti kepada penulis. Segala kasih sayang, motivasi, dan kepercayaan yang diberikan menjadi kekuatan bagi penulis dalam menghadapi setiap proses dan tantangan selama penyusunan skripsi ini. Penulis juga mengucapkan terima kasih kepada adik yang selalu memberikan semangat, dukungan, dan keceriaan, sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik.

**HALAMAN MOTO**

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

(QS. Al-Baqarah: 286)

*“Long story short, I survived.”*

(Taylor Swift)

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan tugas akhir yang berjudul “Pengembangan Aplikasi Tempat Sampah Cerdas Berbasis *Mobile*” dengan baik. Laporan tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana S-1 pada Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Dalam proses penyusunan tugas akhir ini, penulis menyadari bahwa terdapat berbagai kendala yang dihadapi, baik dalam proses pengembangan sistem maupun dalam penyusunan laporan. Namun demikian, berkat bantuan, bimbingan, dan dukungan dari berbagai pihak, laporan ini dapat diselesaikan. Oleh karena itu, melalui kata pengantar ini penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Kedua orang tua penulis yang senantiasa memberi dukungan yang luar biasa kepada penulis baik secara moral dan material. Segala kasih sayang, motivasi dan doa yang tidak pernah berhenti terucap pada setiap langkah penulis. Serta adik penulis Arfa Hadi Firnanda yang selalu memberikan semangat kepada penulis dalam berbagai situasi.
2. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika Fakultas Teknologi Industri.
3. Bapak Dthomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika – Program Sarjana.
4. Bapak IR. Kurniawan Dwi Irianto, S.Kom., M.Sc., selaku dosen pembimbing yang telah memberikan bimbingan, dukungan, saran, dan perhatian dalam penelitian dan penyusunan tugas akhir.
5. Seluruh dosen Program Studi Informatika yang telah memberikan ilmu kepada saya selama proses penyusunan tugas akhir.
6. Keluarga besar yang senantiasa memberikan doa dan dukungan sepanjang proses studi.
7. Bogel lovers yaitu, Alifah, Sekar, Widya, Keysha, Nasywa, Keysa, dan Raisha teman sejak semester satu yang telah memberikan dukungan, semangat, serta kebersamaan dalam menemani suka duka perkuliahan hingga penyusunan penelitian.
8. Rastra, Dendy, Hasan, Gana, Alfian, Raden, Rangga, Naufal, Ayu, Safinatun, dan Novera yang sudah menjadi keluarga kedua saat diperantauan.

9. Kak Tian Nugraha Pratama dan Kak Jefrian Arya Hernanda selaku rekan kolaborasi yang telah memberikan kontribusi, bantuan, serta dukungan selama proses pengerjaan tugas akhir ini.
10. Teman-teman seperjuangan di Program Studi Informatika yang telah memberikan dukungan, bantuan, serta kebersamaan selama proses penyusunan tugas akhir ini.

Penulis tentu menyadari bahwa tugas akhir ini masih jauh dari sempurna. Oleh karena itu, saran dan masukan yang membangun sangat diharapkan untuk penyempurnaan laporan ini. Semoga tugas akhir penulis dapat bermanfaat dan berguna bagi orang lain.

Yogyakarta, 24 April 2026



( Adelia Nur Pratiwi )

## SARI

Permasalahan pengelolaan sampah di lingkungan kampus masih sering terjadi karena proses pemantauan kapasitas tempat sampah masih dilakukan secara manual, sehingga petugas kebersihan harus memeriksa tempat sampah secara langsung ke setiap lokasi. Penelitian ini bertujuan untuk mengembangkan aplikasi tempat sampah cerdas berbasis *mobile* yang dapat membantu petugas kebersihan dalam memantau kondisi tempat sampah secara lebih efektif dan efisien. Metode yang digunakan dalam penelitian ini adalah metode *prototyping* yang meliputi tahap *communication*, *quick plan*, *modeling quick design*, *construction of prototype*, serta *deployment delivery and feedback*. Aplikasi dikembangkan menggunakan Flutter dan Firebase Realtime Database serta terintegrasi dengan perangkat *Internet of Things* (IoT) untuk menampilkan informasi kapasitas tempat sampah secara *real-time*. Fitur utama aplikasi meliputi *login* dan registrasi, *dashboard* monitoring, notifikasi, riwayat aktivitas, akun pengguna, serta penambahan data tempat sampah. Pengujian dilakukan menggunakan *Black Box Testing* dan *System Usability Scale* (SUS). Hasil pengujian menunjukkan bahwa seluruh fitur utama aplikasi berjalan sesuai dengan skenario yang ditentukan, sedangkan hasil pengujian SUS memperoleh skor rata-rata sebesar 76,5 yang termasuk dalam kategori *Good*. Selain itu, hasil observasi pengujian menunjukkan tingkat keberhasilan sistem sebesar 83,3% dalam mendeteksi dan menampilkan informasi sampah melalui aplikasi *mobile*. Dengan demikian, aplikasi tempat sampah cerdas berbasis *mobile* ini dapat membantu proses monitoring tempat sampah dan mendukung pengelolaan kebersihan kampus secara lebih efektif.

Kata kunci: aplikasi *mobile*, tempat sampah cerdas, *Internet of Things*, *Firebase Realtime Database*, *prototyping*, *System Usability Scale*.

## GLOSARIUM

|                               |                                                                                                                                                                 |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Black Box Testing</i>      | metode pengujian perangkat lunak yang berfokus pada input dan output tanpa melihat kode program.                                                                |
| <i>System Usability Scale</i> | metode evaluasi untuk mengukur kemudahan penggunaan sistem melalui kuesioner.                                                                                   |
| <i>Prototyping</i>            | metode pengembangan perangkat lunak dengan membuat model awal sistem untuk diuji dan diperbaiki.                                                                |
| <i>Path Sensor</i>            | alamat data sensor pada Firebase yang digunakan aplikasi untuk membaca data perangkat IoT.                                                                      |
| <i>Internet of Things</i>     | konsep teknologi yang menghubungkan perangkat fisik dengan internet agar dapat mengirim dan menerima data.                                                      |
| <i>Real-time</i>              | kondisi ketika data diperbarui dan ditampilkan secara langsung saat terjadi perubahan.                                                                          |
| <i>Backend as a Service</i>   | layanan <i>backend</i> siap pakai yang membantu pengembang mengelola autentikasi, <i>database</i> , dan sinkronisasi data tanpa harus membangun server sendiri. |
| <i>Firebase</i>               | platform layanan <i>backend</i> dari Google yang digunakan untuk autentikasi, penyimpanan data, dan sinkronisasi data aplikasi secara <i>real-time</i> .        |
| <i>Flutter</i>                | <i>framework open-source</i> dari Google yang digunakan untuk membangun aplikasi <i>mobile</i> dengan satu basis kode.                                          |
| JSON                          | format penyimpanan dan pertukaran data yang tersusun dalam bentuk pasangan nama dan nilai, sering digunakan pada <i>Firebase Realtime Database</i> .            |
| <i>Usability</i>              | tingkat kemudahan suatu sistem atau aplikasi ketika digunakan oleh pengguna untuk mencapai tujuan tertentu.                                                     |
| <i>Wireframe</i>              | rancangan awal tampilan aplikasi yang menggambarkan susunan halaman, komponen, dan alur antarmuka sebelum desain akhir dibuat.                                  |
| <i>Mikrokontroler</i>         | perangkat kecil yang berfungsi untuk memproses data dari sensor dan mengirimkannya ke sistem atau <i>database</i> .                                             |

## 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 .....                                                               | xiii |
| DAFTAR GAMBAR.....                                                               | xiv  |
| BAB I PENDAHULUAN .....                                                          | 1    |
| 1.1 Latar Belakang .....                                                         | 1    |
| 1.2 Rumusan Masalah .....                                                        | 3    |
| 1.3 Batasan Masalah .....                                                        | 3    |
| 1.4 Tujuan Penelitian .....                                                      | 3    |
| 1.5 Manfaat Penelitian .....                                                     | 4    |
| 1.6 Metode Penelitian .....                                                      | 4    |
| 1.7 Sistematika Penulisan .....                                                  | 4    |
| BAB II KAJIAN PUSTAKA .....                                                      | 6    |
| 2.1 Sampah.....                                                                  | 6    |
| 2.2 Pemilahan Sampah dalam Pengelolaan Sampah.....                               | 6    |
| 2.3 Teknologi dan <i>Internet of Things</i> (IoT) dalam Pengelolaan Sampah ..... | 7    |
| 2.4 <i>Software Development</i> .....                                            | 8    |
| 2.5 Tabel Perbandingan.....                                                      | 9    |
| 2.6 Metode <i>Prototyping</i> .....                                              | 10   |
| 2.7 Aplikasi <i>Mobile</i> .....                                                 | 11   |
| 2.8 Flutter .....                                                                | 11   |
| 2.9 Firebase .....                                                               | 12   |
| 2.10 <i>Black Box Testing</i> .....                                              | 13   |
| 2.11 <i>Usability</i> .....                                                      | 13   |
| 2.12 <i>System Usability Scale</i> (SUS).....                                    | 14   |
| 2.13 Penelitian Sejenis .....                                                    | 15   |
| BAB III METODOLOGI.....                                                          | 18   |
| 3.1 <i>Communication</i> (Komunikasi).....                                       | 19   |
| 3.2 <i>Quick Plan</i> (Perencanaan Cepat) .....                                  | 26   |
| 3.3 <i>Modeling Quick Design</i> (Desain Cepat) .....                            | 28   |
| 3.3.1 <i>Use Case Diagram</i> .....                                              | 29   |
| 3.3.2 <i>Activity Diagram</i> .....                                              | 30   |
| 3.3.3 <i>Entity Relationship Diagram</i> (ERD) .....                             | 37   |
| 3.3.4 Desain <i>Wireframe</i> .....                                              | 40   |
| 3.4 <i>Construction of Prototype</i> (Pembangunan Prototipe).....                | 46   |
| 3.5 <i>Deployment Delivery &amp; Feedback</i> (Penerapan dan Umpan Balik) .....  | 49   |
| 3.5.1 Pengujian <i>Black Box Testing</i> .....                                   | 49   |
| 3.5.2 Pengujian <i>System Usability Scale</i> (SUS).....                         | 50   |
| BAB IV HASIL dan PEMBAHASAN .....                                                | 53   |

|       |                                                           |    |
|-------|-----------------------------------------------------------|----|
| 4.1   | Implementasi Sistem .....                                 | 53 |
| 4.1.1 | Halaman Login dan Registrasi .....                        | 53 |
| 4.1.2 | Halaman Dashboard .....                                   | 54 |
| 4.1.3 | Halaman Notifikasi.....                                   | 55 |
| 4.1.4 | Halaman Riwayat .....                                     | 56 |
| 4.1.5 | Halaman Pengaturan.....                                   | 57 |
| 4.1.6 | Halaman Tambah Tempat Sampah .....                        | 58 |
| 4.2   | Integrasi dengan <i>Internet of Things</i> (IoT) .....    | 59 |
| 4.3   | Hasil Pengujian .....                                     | 60 |
| 4.3.1 | Hasil Pengujian <i>Black Box Testing</i> .....            | 60 |
| 4.3.2 | Hasil Pengujian <i>System Usability Scale</i> (SUS) ..... | 61 |
| 4.3.3 | Hasil Observasi Pengujian.....                            | 63 |
| 4.4   | Pembahasan.....                                           | 65 |
|       | BAB V KESIMPULAN dan SARAN .....                          | 67 |
| 5.1   | Kesimpulan .....                                          | 67 |
| 5.2   | Saran.....                                                | 67 |
|       | DAFTAR PUSTAKA .....                                      | 69 |
|       | LAMPIRAN .....                                            | 74 |

## DAFTAR TABEL

|                                                                           |    |
|---------------------------------------------------------------------------|----|
| Tabel 2.1 Perbandingan Metode .....                                       | 9  |
| Tabel 2.2 Penelitian Sejenis.....                                         | 15 |
| Tabel 3.1 Daftar Pertanyaan Wawancara.....                                | 20 |
| Tabel 3.2 Hasil Wawancara .....                                           | 21 |
| Tabel 3.3 Pengujian <i>Black Box Testing</i> .....                        | 49 |
| Tabel 3.4 Pengujian <i>System Usability Scale</i> (SUS).....              | 50 |
| Tabel 3.5 Pertanyaan <i>System Usability Scale</i> (SUS) .....            | 51 |
| Tabel 4.1 Hasil Pengujian Black Box Testing .....                         | 60 |
| Tabel 4.2 Hasil Pengujian <i>System Usability Scale</i> (SUS).....        | 62 |
| Tabel 4.3 Hasil Akhir Pengujian <i>System Usability Scale</i> (SUS) ..... | 63 |

## DAFTAR GAMBAR

|                                                                 |    |
|-----------------------------------------------------------------|----|
| Gambar 3.1 Alur <i>Prototyping</i> .....                        | 18 |
| Gambar 3.2 <i>Use Case Diagram</i> .....                        | 29 |
| Gambar 3.3 <i>Activity Diagram Login</i> .....                  | 31 |
| Gambar 3.4 <i>Activity Diagram Dashboard</i> .....              | 32 |
| Gambar 3.5 <i>Activity Diagram Notifikasi</i> .....             | 34 |
| Gambar 3.6 <i>Activity Diagram Riwayat</i> .....                | 35 |
| Gambar 3.7 <i>Activity Diagram Tambah Tempat Sampah</i> .....   | 36 |
| Gambar 3.8 <i>Entity Relationship Diagram (ERD)</i> .....       | 37 |
| Gambar 3.9 <i>Wireframe Login dan Registrasi</i> .....          | 41 |
| Gambar 3.10 <i>Wireframe Dashboard</i> .....                    | 42 |
| Gambar 3.11 <i>Wireframe Notifikasi</i> .....                   | 43 |
| Gambar 3.12 <i>Wireframe Pengaturan</i> .....                   | 44 |
| Gambar 3.13 <i>Wireframe Riwayat</i> .....                      | 45 |
| Gambar 3.14 Rumus Skor SUS .....                                | 51 |
| Gambar 3.15 Skor SUS.....                                       | 52 |
| Gambar 4.1 Halaman <i>Login dan Registrasi</i> .....            | 54 |
| Gambar 4.2 Halaman <i>Dashboard</i> .....                       | 55 |
| Gambar 4.3 Halaman Notifikasi .....                             | 56 |
| Gambar 4.4 Halaman Riwayat.....                                 | 57 |
| Gambar 4.5 Halaman Pengaturan .....                             | 58 |
| Gambar 4.6 Halaman Tambah Tempat Sampah .....                   | 59 |
| Gambar 4.7 Pengujian Bersama Pengguna.....                      | 62 |
| Gambar 4.8 Hasil Skor <i>System Usability Scale (SUS)</i> ..... | 63 |

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Permasalahan sampah hingga saat ini masih menjadi isu lingkungan yang krusial, khususnya di lingkungan kampus. Menurut *World Health Organization* (WHO) sampah merupakan barang yang tidak terpakai, tidak dibutuhkan, tidak diinginkan, atau hal-hal yang dibuang yang berasal dari aktivitas manusia dan bukan hasil dari proses alami (Marlina et al., 2023). Menurut Undang-undang No.18 Tahun 2008 tentang pengelolaan sampah, sampah merupakan hasil dari aktivitas sehari-hari manusia atau proses alami yang berwujud padat atau semi padat, terdiri dari bahan organik maupun anorganik yang bisa terurai atau tidak terurai, dan dianggap tidak bermanfaat lagi sehingga dibuang ke lingkungan (Fiqih & Syaiful, 2023). Peneliti sebelumnya telah menunjukkan betapa pentingnya menggunakan teknologi informasi dalam pengelolaan sampah dan lingkungan hidup (Dzakiyyudin & Frendiana, 2024). Teknologi dinilai mampu memberikan solusi dalam meningkatkan efektivitas dan efisiensi pengelolaan sampah, khususnya di lingkungan kampus yang memiliki aktivitas tinggi.

Menurut penelitian yang dilakukan oleh Bakhtiar et al. (2023) di kantin Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang, ada keluhan tentang tumpukan sampah yang tidak langsung terangkut meskipun sistem pengelolaan sampah organik sudah ada. Kondisi ini menimbulkan banyak keluhan dari pengunjung kantin dan juga mempengaruhi kebersihan dan kesehatan lingkungan sekitar. Selain itu, minimnya kesadaran mengenai pentingnya pemilahan sampah oleh civitas akademik juga salah satu penyebab permasalahan tersebut. Program pengelolaan sampah yang telah diterapkan pun belum sepenuhnya efektif dalam pengelolaan penumpukan sampah. Oleh karena itu, perlu dilakukan evaluasi dan perbaikan sistem pengelolaan sampah yang lebih optimal di area kampus.

Permasalahan serupa juga ditemukan di lingkungan Fakultas Teknologi Industri Universitas Islam Indonesia. Berdasarkan hasil observasi dan wawancara dengan petugas kebersihan serta mahasiswa, diketahui bahwa sistem monitoring tempat sampah masih dilakukan secara manual. Petugas kebersihan harus memeriksa tempat sampah yang berada di tiap lantai gedung untuk memastikan bahwa tempat sampah tidak dalam keadaan penuh. Hal ini sangat tidak efisien dan menyulitkan petugas kebersihan dalam pengelolaan waktu. Selain itu, tempat sampah yang ada di area Fakultas juga masih menggunakan satu jenis tanpa pemisahan antara sampah organik, anorganik, dan juga B3 (Bahan Berbahaya dan Beracun). Kondisi ini menyulitkan petugas kebersihan karena semua sampah yang masih tercampur

menjadi satu, sehingga petugas kebersihan harus memilah sampah terlebih dahulu sebelum melakukan proses pembuangan lanjutan.

Penelitian lainnya yang dilakukan oleh Sekti, Ulfa, and Anwar (2024) di Universitas Esa Unggul Tangerang juga menemukan masalah serupa. Di kampus, tempat sampah belum dilengkapi dengan sistem pemantauan otomatis, sehingga pengecekan kapasitas tempat sampah harus dilakukan secara manual. Hal ini menyulitkan petugas kebersihan karena mereka harus menghabiskan waktu dan tenaga lebih banyak untuk memantau kondisi tempat sampah di area kampus. Selain itu, tempat sampah yang masih belum dibedakan berdasarkan jenisnya, sehingga sampah organik, anorganik, dan B3 tercampur menjadi satu. Situasi seperti ini dapat menyebabkan penumpukan sampah dan mengurangi efektivitas program kebersihan kampus. Oleh karena itu, dibutuhkan sistem pengelolaan sampah yang lebih canggih dengan teknologi yang dapat membantu memantau kapasitas tempat sampah dan secara otomatis memisahkan jenis sampah.

Untuk mengatasi masalah tersebut, salah satu cara yang bisa dilakukan adalah pengembangan aplikasi *mobile* yang terintegrasi dengan sistem pengelolaan tempat sampah cerdas. Aplikasi ini dibuat untuk membantu petugas kebersihan dalam memantau keadaan kapasitas tempat sampah di area Fakultas secara langsung melalui data yang dikirim dari sensor yang terpasang pada tempat sampah. Selain itu, aplikasi ini juga akan memberikan notifikasi ketika kapasitas tempat sampah sudah penuh, sehingga petugas bisa segera melakukan pengangkutan sampah tanpa melakukan pemantauan manual. Dengan adanya aplikasi ini pengelolaan sampah di lingkungan Fakultas diharapkan menjadi lebih teratur dan efisien. Sistem ini diharapkan dapat membantu meningkatkan produktivitas petugas kebersihan dan mendukung terciptanya lingkungan yang bersih dan nyaman.

Penelitian yang dilakukan sebelumnya oleh Wijaya et al. (2025) telah menciptakan sistem tempat sampah pintar yang menggunakan teknologi *Internet of Things* (IoT) untuk secara otomatis mendeteksi kapasitas limbah dan mengirimkan pemberitahuan melalui SMS. Meskipun demikian, sistem itu masih belum terhubung dengan aplikasi *mobile* yang menyediakan tampilan visual secara langsung, rekaman aktivitas, serta pemetaan lokasi tempat sampah. Di samping itu, belum ada fitur pendidikan atau keterlibatan yang lebih luas bagi pengguna di lingkungan kampus. Di sisi lain, penelitian oleh Fitriana, Hashina, and Tanjung (2022) memfokuskan pada pembuatan aplikasi Android untuk mencatat transaksi bank sampah, tetapi belum memiliki dukungan untuk integrasi dengan alat sensor dan tidak dapat memonitor kapasitas sampah secara otomatis. Aplikasi ini juga hanya diperuntukkan bagi

pengguna internal (petugas KSM) dan belum mencakup partisipasi dari civitas akademika secara keseluruhan.

Berdasarkan uraian permasalahan yang telah dipaparkan, dapat disimpulkan bahwa pengelolaan sampah di area kampus masih mengalami berbagai kendala. Hal ini mulai dari sistem pemantauan kapasitas tempat sampah yang masih manual serta belum adanya penerapan pemilahan jenis sampah secara langsung. Oleh karena itu, penelitian ini bertujuan untuk menciptakan aplikasi tempat sampah cerdas berbasis *mobile* yang mampu memantau kondisi tempat sampah secara *real-time* dan pemilahan jenis sampah. Diharapkan aplikasi ini bisa bermanfaat dengan membantu petugas kebersihan serta menjadi solusi untuk mendukung kebersihan lingkungan kampus.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah pada penelitian ini, yaitu: Bagaimana merancang dan mengembangkan aplikasi tempat sampah cerdas berbasis *mobile* di lingkungan Fakultas Teknologi Industri Universitas Islam Indonesia yang dapat membantu petugas kebersihan dalam monitoring tempat sampah dan pemilahan jenis sampah.

## 1.3 Batasan Masalah

Dalam penelitian ini, peneliti akan membatasi ruang lingkup pembahasannya agar lebih terarah dan fokus, maka ditetapkan batasan masalah sebagai berikut:

- a. Aplikasi yang dikembangkan hanya berbasis *mobile* Android.
- b. Lokasi tempat sampah yang digunakan sebagai objek penelitian terbatas pada area Fakultas Teknologi Industri Universitas Islam Indonesia.
- c. Sistem monitoring hanya meliputi kapasitas tempat sampah dan pemilahan sampah jenis organik, botol plastik, dan kaleng.
- d. Aplikasi hanya digunakan oleh petugas kebersihan Fakultas Teknologi Industri.
- e. Pengembangan aplikasi dilakukan menggunakan Bahasa pemrograman Flutter dan Firebase untuk databasenya.
- f. Pemantauan lokasi tempat sampah dilakukan secara statis, tanpa memanfaatkan sistem pelacakan yang menggunakan GPS atau penggabungan dengan peta digital yang bergerak.

## 1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk merancang dan mengembangkan aplikasi tempat sampah cerdas berbasis *mobile* yang dapat membantu petugas kebersihan di area Fakultas

Teknologi Industri Universitas Islam Indonesia dalam memantau kapasitas tempat sampah dan pemilahan jenis sampah.

### 1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

- a. Untuk petugas kebersihan, membantu petugas kebersihan dalam memantau kondisi tempat sampah tanpa keliling untuk memeriksa langsung, sehingga dapat menghemat waktu dan tenaga.
- b. Untuk penulis, dapat mengembangkan dan menerapkan ilmu yang diperoleh selama kuliah, khususnya metode dan teknologi dalam pengembangan perangkat lunak berbasis Android.

### 1.6 Metode Penelitian

Penelitian ini menggunakan metode *prototyping* sebagai pendekatan dalam proses pengembangan aplikasi. Adapun beberapa Langkah-langkah penelitian yang dapat dilakukan melalui tahapan berikut:

#### a. *Communication*

Proses pengumpulan data melalui observasi dan wawancara untuk mengetahui permasalahan pengelolaan tempat sampah serta kebutuhan pengguna.

#### b. *Quick Plan*

Menyusun kebutuhan fungsional dan non-fungsional sistem berdasarkan hasil pengumpulan data.

#### c. *Modeling Quick Design*

Penyusunan rancangan sistem berupa *use case diagram*, *activity diagram*, *Entity Relationship Diagram* (ERD), dan rancangan antarmuka aplikasi.

#### d. *Construction of Prototype*

Prototipe dikembangkan dengan memperhatikan fitur utama sistem, seperti autentikasi pengguna, pemantauan kapasitas tempat sampah, notifikasi, riwayat aktivitas, serta penambahan data tempat sampah.

#### e. *Deployment and Feedback*

Penerapan dan pengujian aplikasi kepada pengguna.

### 1.7 Sistematika Penulisan

Sistematika penulisan dalam penyusunan laporan tugas akhir adalah sebagai berikut:

- a. BAB I PENDAHULUAN

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

b. BAB II KAJIAN PUSTAKA

Bab ini berisi kajian literatur dan teori-teori dasar yang mendukung penelitian ini.

c. BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang metode yang digunakan dalam pengembangan aplikasi, tahapan penelitian, membangun desain sistem dan *prototype*, pengujian dan evaluasi *prototype*.

d. BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi hasil pembahasan dari pengembangan aplikasi dan pengujian yang sudah dilakukan.

e. BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil penelitian dan pembahasan yang telah dilakukan. Selain itu juga terdapat saran untuk pengembangan sistem atau penelitian selanjutnya.

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1 Sampah**

Menurut *World Health Organization* (WHO) sampah merupakan sesuatu yang tidak dipakai, tidak digunakan, tidak diinginkan, atau sesuatu yang dibuang yang berasal dari kegiatan manusia dan tidak terjadi secara alam (Marlina et al., 2023). Menurut UU No. 18 Tahun 2008 tentang pengelolaan sampah merupakan hasil dari aktivitas harian manusia dan/atau proses alami yang memiliki bentuk fisik.

Sampah berdasarkan jenisnya dibagi menjadi tiga, yaitu:

- a. Sampah organik adalah jenis sampah yang dihasilkan dari sisa makanan atau bahan organik lainnya. Karena itu, sampah organik dapat terurai lebih cepat melalui pembusukan yang dibantu oleh bakteri dibandingkan dengan sampah anorganik (Marlina et al., 2023).
- b. Sampah anorganik adalah sampah yang tidak digunakan lagi dan sulit untuk terurai secara alami yang berasal dari bahan sintesis seperti plastik, kaca, atau logam (Olethea Farah Utari et al., 2024).
- c. Sampah B3 (Bahan Berbahaya dan Beracun) adalah sampah rumah tangga yang memiliki unsur berbahaya dan beracun yang dihasilkan oleh aktivitas masyarakat di area tempat tinggal yang dapat menimbulkan ancaman bagi lingkungan dan kesehatan masyarakat (Marlina et al., 2023).

Pengelolaan sampah yang baik sangat penting untuk menjaga lingkungan bersih dan mencegah dampak buruk terhadap lingkungan, terutama di lingkungan fasilitas publik seperti kampus.

#### **2.2 Pemilahan Sampah dalam Pengelolaan Sampah**

Salah satu prinsip utama dalam pengelolaan sampah adalah pemilahan sampah berdasarkan jenis, seperti sampah organik, plastik, dan kaleng. Pemilahan ini memungkinkan proses pengangkutan dan pengolahan sampah dilakukan secara lebih terarah sesuai dengan karakteristik masing-masing jenis sampah, sehingga dapat meningkatkan efisiensi penanganan dan mengurangi pencampuran sampah sejak dari sumbernya. Studi menunjukkan bahwa pengelolaan sampah tanpa pemilahan cenderung meningkatkan beban kerja petugas dan menyulitkan proses pengolahan lanjutan (Olethea Farah Utari et al., 2024).

Pemilahan sampah menjadi semakin penting di lingkungan kampus karena volume sampah yang dihasilkan beragam dan berasal dari berbagai aktivitas akademik. Menurut Bakhtiar et al. (2023), penerapan sistem pemilahan sampah di fasilitas publik dapat membantu petugas kebersihan menentukan jenis sampah mana yang paling penting untuk ditangani dan mengurangi risiko penumpukan sampah tertentu, terutama sampah plastik yang memiliki volume yang besar dan sulit terurai secara alami.

Perangkat *Internet of Things* (IoT) yang terpasang di tempat sampah dapat mengirimkan data kapasitas ke sistem secara *real-time* dan menampilkan jenis sampah melalui aplikasi digital dan sistem berbasis sensor, memungkinkan proses pemilahan sampah menjadi lebih mudah daripada sebelumnya. Terbukti bahwa metode ini dapat membantu dalam pengelolaan sampah yang lebih sistematis dan meningkatkan efisiensi pemantauan kondisi tempat sampah (Sekti et al., 2024).

Kombinasi teknologi digital dan pemilahan sampah membantu petugas kebersihan dalam kegiatan operasional sekaligus mendukung aspek edukasi lingkungan. Aplikasi *mobile* dapat menampilkan informasi tentang jenis dan kondisi sampah untuk meningkatkan kesadaran pengguna tentang pentingnya pemilahan sampah. Oleh karena itu, landasan konseptual untuk penelitian ini adalah pemilahan sampah berbasis teknologi. Sistem tempat sampah cerdas ini akan menggunakan kemampuan perangkat IoT untuk membedakan sampah organik, plastik, dan kaleng serta menyajikan informasi dalam format yang mudah dipahami.

Dalam penelitian ini, konsep pemilahan sampah diterapkan sesuai dengan kemampuan perangkat IoT yang digunakan. Sistem dirancang untuk membedakan tiga jenis sampah utama, yaitu sampah organik, plastik, dan kaleng, sehingga pemantauan kapasitas dan penyajian informasi pada aplikasi *mobile* disesuaikan dengan klasifikasi tersebut.

### **2.3 Teknologi dan *Internet of Things* (IoT) dalam Pengelolaan Sampah**

Perkembangan teknologi digital telah mengubah banyak hal, termasuk manajemen sampah. Penggunaan perangkat digital seperti sensor, mikrokontroler, dan sistem monitoring berbasis jaringan memungkinkan proses pemantauan dan pengangkutan sampah secara otomatis, efisien, dan *real-time*. Ini karena sistem konvensional yang dilakukan secara manual dianggap kurang efisien, tidak responsif terhadap kondisi aktual, dan memerlukan banyak tenaga dan waktu (Agustino et al., 2024).

*Internet of Things* (IoT) adalah salah satu teknologi yang paling relevan untuk pengelolaan sampah kontemporer karena memungkinkan objek fisik, seperti tempat sampah,

dilengkapi dengan sensor dan terhubung ke jaringan internet. Ketinggian sampah dalam wadah diukur oleh sensor ultrasonik, yang kemudian diproses oleh mikrokontroler dan dikirim melalui jaringan nirkabel (Bakhtiar et al., 2023). Perangkat ini dapat mengirimkan data kapasitas sampah terus menerus ke sistem pemantauan, yang dapat diakses oleh petugas melalui aplikasi *mobile*.

## 2.4 Software Development

*Software development* adalah aktivitas yang terjadi dalam pengembangan perangkat lunak. Dengan kata lain, pengembangan perangkat lunak adalah proses pembuatan, perancangan, pengujian, dan pemeliharaan perangkat lunak komputer. Semua tindakan ini dilakukan oleh pengembang perangkat lunak, baik secara individu maupun dalam kelompok, untuk menghasilkan produk perangkat lunak yang baik dan sesuai dengan kebutuhan pengguna (Maulana et al., 2024).

*Software development* tidak hanya berfokus pada penulisan kode program, tetapi juga mencakup perencanaan, analisis kebutuhan pengguna, perancangan sistem, pengujian, dan evaluasi hasil. Semua rangkaian ini penting untuk memastikan perangkat lunak yang dibuat dapat berfungsi secara optimal dan mudah digunakan.

Selain itu, proses pengembangan perangkat lunak juga didefinisikan sebagai rangkaian tindakan yang bertujuan untuk membuat, mengubah, meningkatkan, dan memelihara perangkat lunak. Menurut Yas, Alazzawi, and Rahmatullah (2023), proses pengembangan perangkat lunak biasanya terdiri dari beberapa langkah berurutan yang menjelaskan bagaimana perangkat lunak dibuat, diuji, dan dipelihara untuk mencapai kualitas yang diharapkan pengguna.

*Software development* melibatkan aspek teknis selain komunikasi dan kerja sama pengembang untuk menyelesaikan masalah pengguna. *Software development* dapat didefinisikan sebagai proses rekayasa yang mengintegrasikan elemen teknis, analitis, dan kolaboratif untuk menghasilkan perangkat lunak yang efektif dan memberikan manfaat nyata bagi pengguna. Ini karena pengembang harus memahami konteks penggunaan perangkat lunak dan menyesuaikannya dengan kebutuhan yang ada. *Software development* digunakan sebagai landasan dalam mengembangkan aplikasi *mobile* yang terintegrasi, sehingga proses pengembangan sistem dapat dilakukan secara terstruktur dan terarah.

## 2.5 Tabel Perbandingan

Tabel 2.1 Perbandingan Metode

| Metode                      | Kelebihan Utama                                                                | Kelemahan Utama                                                | Cocok untuk <i>Mobile</i> | Dukungan Iteratif | Keterlibatan Pengguna | Kecepatan Pengembangan |
|-----------------------------|--------------------------------------------------------------------------------|----------------------------------------------------------------|---------------------------|-------------------|-----------------------|------------------------|
| <i>Waterfall</i>            | Tahapan jelas, dokumentasi lengkap                                             | Tidak fleksibel terhadap perubahan kebutuhan                   | Kurang                    | Tidak             | Rendah                | Sedang                 |
| <i>Design and Creation</i>  | Fokus langsung ke pembuatan produk nyata                                       | Minim dokumentasi, kurang cocok untuk sistem besar             | Ya                        | Rendah            | Sedang                | Cepat                  |
| <i>User Centered Design</i> | Sangat sesuai kebutuhan pengguna, melibatkan user sejak awal                   | Butuh waktu lebih banyak di tahap riset dan validasi kebutuhan | Ya                        | Tinggi            | Tinggi                | Lambat                 |
| <i>Prototyping</i>          | Iteratif, fleksibel, mudah diuji coba dan direvisi berdasarkan <i>feedback</i> | Butuh waktu lama jika banyak iterasi                           | Sangat cocok              | Tinggi            | Tinggi                | Cepat                  |
| <i>Agile</i>                | Sangat fleksibel, bisa menyesuaikan kebutuhan di tengah pengembangan           | Butuh tim kecil yang solid dan disiplin                        | Sangat cocok              | Sangat tinggi     | Tinggi                | Cepat                  |
| <i>Scrum</i>                | Manajemen progress per                                                         | Butuh <i>scrum</i> master kompeten,                            | Sangat cocok              | Sangat tinggi     | Tinggi                | Cepat                  |

|  |                           |                                      |  |  |  |  |
|--|---------------------------|--------------------------------------|--|--|--|--|
|  | sprint, iteratif, adaptif | jadwal <i>sprint</i> berat untuk tim |  |  |  |  |
|--|---------------------------|--------------------------------------|--|--|--|--|

Berdasarkan tabel di atas, dapat dilihat bahwa metode *Prototyping* memiliki kelebihan dalam hal fleksibilitas, kemampuan iterasi, kecepatan dalam pengembangan, serta tingkat partisipasi pengguna yang tinggi. Ini sangat cocok dengan kebutuhan proyek pengembangan aplikasi tempat sampah cerdas berbasis *mobile* di area kampus, yang memerlukan uji *prototype* yang cepat dan perbaikan berulang berdasarkan *feedback* dari pengguna.

Di samping itu, *prototyping* juga memberikan kesempatan untuk menyesuaikan fitur selama proses pengembangan berlangsung tanpa perlu memulai dari nol seperti yang terjadi pada metode *waterfall*. Dengan demikian, metode *prototyping* dipilih sebagai pendekatan pengembangan sistem dalam penelitian ini karena dianggap paling sesuai dengan karakteristik proyek serta keterbatasan waktu dan sumber daya yang ada di lingkungan kampus.

## 2.6 Metode *Prototyping*

Metode *prototyping* adalah teknik pengembangan perangkat lunak yang menekankan pada pembuatan model awal (*mockup*) dari sistem yang akan dibangun. *Prototype* dimaksudkan untuk memberikan gambaran awal tentang fungsi, alur kerja, dan tampilan sistem sehingga pengembang dan pengguna memiliki pemahaman yang sama tentang sistem yang akan dibangun. Dengan menggunakan *prototype*, kebutuhan pengguna dapat diidentifikasi dan dipahami dengan lebih baik (Fridayanthie et al., 2021).

Dalam metode *prototyping*, proses pengembangan dimulai dengan pengumpulan kebutuhan awal yang masih bersifat umum. Berdasarkan kebutuhan tersebut, pengembang membuat *prototype* sistem dalam bentuk model awal atau *mockup*. Kemudian, pengguna mengevaluasi *prototype* untuk mengetahui apakah fungsi dan tampilannya sesuai atau tidak. Hasil evaluasi ini digunakan sebagai dasar untuk melakukan perbaikan dan penyempurnaan *prototype* secara berulang hingga rancangan sistem dinilai telah memenuhi kebutuhan pengguna.

Metode *prototyping* mengutamakan partisipasi pengguna selama proses pengembangan. Dengan keterlibatan ini, pengguna dapat secara langsung memberikan masukan ke sistem yang dirancang, sehingga perubahan yang diperlukan dapat diakomodasi sejak awal tanpa harus mengulangi keseluruhan proses pengembangan. Metode ini meningkatkan kesesuaian sistem

dengan kebutuhan pengguna dan mengurangi kemungkinan kesalahan implementasi (Paksi et al., 2023).

Metode *prototyping* memungkinkan proses pengembangan dilakukan secara lebih fleksibel dan iteratif, serta menghasilkan perangkat lunak yang lebih sesuai dengan tujuan dan kebutuhan pengguna. Selain itu, *prototype* yang telah disetujui oleh pengguna akan dijadikan dasar dalam proses pengkodean, sehingga sistem yang dibangun memiliki alur dan fungsionalitas yang lebih terarah (Alda et al., 2023).

## 2.7 Aplikasi Mobile

Aplikasi *mobile* merupakan perangkat lunak yang dibuat untuk memungkinkan pengguna mengakses informasi dan layanan secara fleksibel, cepat, dan *real-time* tanpa dibatasi oleh ruang dan waktu. Aplikasi ini dapat digunakan pada perangkat bergerak seperti *smartphone* dan tablet. Aplikasi *mobile* memiliki fitur ini, yang menjadikannya salah satu media utama untuk menyediakan layanan digital pada berbagai bidang (Putra et al., 2023).

Aplikasi *mobile* dirancang untuk meningkatkan fungsionalitas dan kemudahan penggunaan (*usability*) dan pengalaman pengguna. Aplikasi *mobile* biasanya memiliki antarmuka pengguna yang mudah digunakan dan mendukung berbagai fitur, seperti notifikasi, pengelolaan data, dan konektivitas jaringan.

Aplikasi *mobile* berfungsi sebagai media utama interaksi antara pengguna dan sistem dalam sistem informasi. Selain memungkinkan pemantauan data secara langsung melalui perangkat pengguna, aplikasi *mobile* memungkinkan proses pengelolaan data dan layanan dilakukan dengan lebih efisien. Oleh karena itu, penggunaan aplikasi *mobile* dianggap dapat meningkatkan efisiensi dan kualitas layanan yang diberikan kepada pengguna (Shabrina Ziha Fidela et al., 2023). Pada penelitian ini, aplikasi *mobile* dimanfaatkan sebagai media utama bagi petugas kebersihan untuk memantau kondisi tempat sampah secara langsung tanpa harus melakukan pengecekan manual di lapangan.

## 2.8 Flutter

Flutter merupakan *framework* yang bersifat *open-source*, yang dikembangkan oleh Google untuk memudahkan pembuatan antarmuka. Flutter dapat digunakan untuk mengembangkan aplikasi *mobile*, web, dan desktop. Untuk pengembangan aplikasi *mobile*, flutter bisa dimanfaatkan dalam membuat aplikasi untuk Android dan iOS. Tujuan dari pendekatan ini yaitu untuk meningkatkan efisiensi pengembangan aplikasi serta mengurangi waktu dan biaya (Awaludin et al., 2023; Setiawan et al., 2025). Flutter mendukung pembuatan

aplikasi dengan pengalaman pengguna yang seragam pada berbagai platform, sehingga banyak digunakan dalam pengembangan aplikasi *mobile* modern. Ini karena Flutter menggunakan pendekatan antarmuka berbasis *widget*, yang memungkinkan pengembang membuat antarmuka yang konsisten, responsif, dan mudah dikustomisasi (Awaludin et al., 2023).

Kemudahan pembelajaran dan fleksibilitas penggunaannya membuat Flutter terus berkembang. Menurut penelitian Setiawan and Wandri (2025), Flutter membantu pengembang membuat aplikasi yang dinamis dan interaktif dan mempercepat proses pengembangan, menjadikannya teknologi yang relevan dan adaptif untuk memenuhi kebutuhan pengembangan aplikasi *mobile* di era digital.

Flutter mendukung proses perancangan dan evaluasi aplikasi secara cepat, yang menjadikannya pilihan yang baik dalam konteks pengembangan aplikasi berbasis *prototyping*. Menurut Mamoun et al., (2021), Flutter dapat digunakan untuk membuat *prototype* aplikasi *mobile* yang dapat diuji secara langsung pada perangkat *smartphone* untuk mengevaluasi fungsi dan antarmuka aplikasi sebelum dilanjutkan ke tahap implementasi penuh. Oleh karena itu, Flutter sangat relevan untuk pengembangan aplikasi secara fleksibel, sehingga sesuai dengan pendekatan *prototyping*.

## 2.9 Firebase

Firebase adalah layanan *Backend as a Service* (BaaS) yang dibuat oleh Google. Firebase membantu para pengembang perangkat lunak dalam menciptakan aplikasi berbasis website dan *mobile*. Layanan *backend* Firebase seperti autentikasi pengguna, basis data, mengirimkan notifikasi, dan sinkronisasi data secara *real-time* menghilangkan kebutuhan pengembang untuk membangun dan mengelola server sendiri (Panjaitan & Pakpahan, 2021)

Firebase Realtime Database adalah layanan utama Firebase. Ini adalah basis data NoSQL yang memungkinkan sinkronisasi data langsung ke semua klien yang terhubung dan menyimpan data dalam format JSON. Firebase cocok untuk aplikasi yang membutuhkan tanggapan data cepat dan interaktif karena setiap perubahan data akan diperbarui secara otomatis tanpa memerlukan proses *refresh* (Khairun Nisa Meiah Ngafidin et al., 2021).

Firebase tidak hanya memiliki Realtime Database, tetapi juga memiliki layanan untuk autentikasi dan penyimpanan data yang terintegrasi. Firebase memudahkan komunikasi aplikasi-server, autentikasi pengguna, dan penyimpanan data, menurut (Panjaitan & Pakpahan, 2021). Dengan Firebase, dapat mempercepat proses pengembangan sistem dengan mengintegrasikan berbagai *library* ke berbagai *framework* pengembangan aplikasi. Firebase

cocok untuk pengembangan aplikasi berbasis *mobile* yang membutuhkan kecepatan dan keandalan sistem karena sinkronisasi data secara *real-time*, kemudahan integrasi, dan layanan *backend* yang siap digunakan.

### **2.10 Black Box Testing**

*Black Box Testing* merupakan metode pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tanpa memeriksa kode program internal. Pengujian dilakukan untuk menentukan serangkaian kondisi *input* dan *output* yang dihasilkan sesuai untuk menguji fungsionalitas aplikasi berjalan sesuai dengan kebutuhan fungsional dan spesifikasi yang telah ditetapkan (Ammann & Offutt, 2016).

*Black Box Testing* dilakukan dari perspektif pengguna dengan menyediakan berbagai skenario penggunaan sistem dan melihat reaksinya. Fokus utama pengujian ini adalah untuk memastikan bahwa fungsi aplikasi seperti autentikasi, pengolahan data, penampilan informasi, notifikasi, dan navigasi antar halaman berjalan dengan baik (Muhammad Helmi Satria Fedianto et al., 2023).

*Black Box Testing* digunakan untuk menemukan kesalahan fungsional, kesalahan antarmuka pengguna, dan ketidaksesuaian antara keluaran sistem dan kebutuhan pengguna. Ini cocok untuk aplikasi berbasis *mobile* karena mampu memeriksa perilaku sistem secara menyeluruh tanpa bergantung pada detail implementasi program (Muhammad Helmi Satria Fedianto et al., 2023). *Black Box Testing* adalah metode yang tepat untuk memastikan bahwa semua fitur utama aplikasi dapat berfungsi dengan baik sebelum sistem digunakan oleh pengguna.

### **2.11 Usability**

*Usability* adalah ukuran sejauh mana suatu sistem atau aplikasi dapat digunakan oleh pengguna untuk mencapai tujuan tertentu secara efektif, efisien, dan memberikan kepuasan dalam penggunaannya. Sebuah aplikasi dianggap *usability* yang baik jika pengguna dapat memahami cara sistem bekerja, menyelesaikan tugas dengan mudah, dan merasa nyaman berinteraksi dengannya.

*Usability* sangat penting untuk keberhasilan aplikasi berbasis *mobile* karena pengguna berinteraksi dengan sistem melalui perangkat dengan ukuran layar terbatas dan pola penggunaan yang cepat. Antarmuka yang rumit, navigasi yang tidak jelas, dan proses yang rumit dapat mengurangi *usability* aplikasi dan menyulitkan pengguna untuk menggunakannya. Oleh karena itu, *usability* sangat penting untuk keberhasilan aplikasi berbasis *mobile*.

Menurut Ernawati & Rahayu (2023), evaluasi *usability* diperlukan untuk menentukan sejauh mana aplikasi mampu mendukung aktivitas pengguna secara optimal. Evaluasi *usability* dapat menemukan masalah pada antarmuka dan fungsi sistem yang dapat menghambat efektivitas, efisiensi, dan kepuasan pengguna. Aplikasi dengan tingkat *usability* yang baik akan membantu pengguna menyelesaikan tugas dengan lebih cepat, mengurangi kesalahan penggunaan, dan meningkatkan pengalaman pengguna.

### 2.12 System Usability Scale (SUS)

Metode yang digunakan adalah *Sistem Usability Scale* (SUS) yaitu evaluasi yang menilai persepsi pengguna terhadap kemudahan penggunaan sistem. Menurut 'Aisy et al., (2024) SUS efektif digunakan untuk mengevaluasi aplikasi *mobile* karena mampu memberikan penilaian kuantitatif yang mudah dipahami. Di sisi lain, penelitian yang dilakukan oleh Ratnawati and Anandito (2024) menunjukkan bahwa SUS memberikan hasil yang konsisten dalam menilai kepuasan dan pengalaman pengguna.

Metode ini menghasilkan skor akhir dalam rentang 0 hingga 100. Pertanyaan dengan nomor ganjil (1, 3, 5, 7, 9) skor jawaban dikurangi 1, sedangkan pertanyaan dengan nomor genap (2, 4, 6, 8, 10), skor 5 dikurangi skor jawaban. Skor total kemudian dijumlahkan dan dikalikan dengan 2,5 ('Aisy et al., 2024).

Menurut interpretasi, hasil tes SUS dibagi menjadi beberapa kategori:

- a. Tingkat Penerimaan: Skor di bawah 50 dianggap "*Not Acceptable*", skor antara 50 dan 70 dianggap "*Marginal*", dan skor di atas 70 dianggap "*Acceptable*".
- b. Menurut skala huruf: Skor 80 hingga 100 dianggap sebagai A, skor 68 hingga 79 dianggap sebagai B, dan skor di bawah 51 dianggap sebagai F.
- c. Deskripsi Penilaian (*Adjective Rating*): Digunakan untuk memberi gambaran subjektif seperti "*Poor*", "*OK*", "*Good*", dan "*Excellent*".

SUS dapat digunakan untuk mengevaluasi aplikasi berbasis *mobile* karena sangat efektif dalam memberikan gambaran kuantitatif tentang pengalaman pengguna secara keseluruhan. Selain itu, SUS memberikan dasar bagi pengembang untuk menentukan area yang membutuhkan perbaikan, sehingga aplikasi dapat dikembangkan sesuai kebutuhan dan ekspektasi pengguna.

### 2.13 Penelitian Sejenis

Penelitian ini mencakup kajian literatur yang diperoleh melalui *Google Scholar* untuk memperkuat penelitian yang dilakukan. Penelitian difokuskan dengan menggunakan kata kunci “Aplikasi tempat sampah cerdas”, “IoT waste monitoring”, dan “aplikasi pengelolaan sampah mobile”.

Beberapa penelitian yang relevan ditampilkan pada tabel:

Tabel 2.2 Penelitian Sejenis

| Peneliti dan Tahun                      | Metode             | Platform              | Judul                                                                                                                                             | Fitur                                                                                                   |
|-----------------------------------------|--------------------|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| Ramdhani, Yustiana, and Fergina (2022)  | <i>Prototyping</i> | <i>Website</i>        | Sistem Informasi Bank Sampah Lembur Sawah, Sukabumi                                                                                               | Manajemen data nasabah, manajemen data sampah, transaksi setor sampah, laporan PDF                      |
| Dzakiyyudin and Frendiana (2024)        | <i>Waterfall</i>   | <i>Mobile</i>         | Perancangan aplikasi nasabah bank sampah di kampung proklamasi untuk memantau intensif ekonomi dan transaksi bank sampah.                         | Dashboard dengan kartu virtual NIK & saldo intensif, menu dompet, integrasi API untuk koneksi database. |
| Kholili, (2023)                         | Analisis sistem    | <i>Mobile Android</i> | Pengembangan sistem informasi jemput sampah rumah tangga berbasis <i>mobile</i> untuk mempermudah koordinasi antara warga dan petugas kebersihan. | Manajemen data user, jenis sampah, transportasi, chat pribadi, monitoring lapangan.                     |
| Ismail, Abdullah, and Abdussamad (2021) | Perancangan Sistem | <i>Web server</i>     | Rancang bangun tempat sampah pintar dengan sistem monitoring jarak jauh berbasis IoT.                                                             | Sensor ultrasonik.                                                                                      |
| Alphita and Saian (2023)                | <i>Waterfall</i>   | <i>Mobile Android</i> | Aplikasi edukasi interaktif tentang pengelolaan sampah                                                                                            | Deteksi sampah, riwayat deteksi & peta lokasi, informasi jenis                                          |

|                                                   |                                      |                             |                                                                                                                                 |                                                                                                                                                  |
|---------------------------------------------------|--------------------------------------|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                   |                                      |                             | menggunakan deteksi gambar berbasis AI                                                                                          | & cara pengolahan sampah                                                                                                                         |
| Fitriana et al. (2022)                            | Kanban                               | Android                     | Pengembangan aplikasi pencatatan transaksi bank sampah desa kalibagor untuk petugas KSM                                         | Input & manajemen data transaksi, penyimpanan data <i>real-time</i>                                                                              |
| Olethea Farah Utari et al. (2024)                 | <i>User Centered Design</i>          | Android                     | Aplikasi DTrash untuk penjemputan sampah organik di perumahan Puri Bojong Lestari.                                              | Penjemputan sampah, edukasi pengelolaan sampah, notifikasi jadwal penjemputan.                                                                   |
| Sarip Hidayatuloh, Nadia Setinanda Pratami (2021) | <i>Rapid Application Development</i> | Web-based                   | Sistem transaksi bank sampah untuk pengelolaan sampah di kampus UIN Jakarta.                                                    | <i>Input &amp; validasi</i> transaksi tabungan sampah, perhitungan ranking nasabah dengan metode SAW, laporan rekap transaksi <i>real-time</i> . |
| Sekti et al. (2024)                               | Rancang Bangun                       | IoT + Blynk App             | <i>Prototype</i> tempat sampah cerdas di Universitas Esa Unggul Tangerang dengan sistem monitoring kapasitas <i>real-time</i> . | Monitoring kapasitas <i>real-time</i> , notifikasi otomatis, DFPlayer mini + speaker untuk notif suara.                                          |
| Wijaya et al. (2025)                              | <i>Waterfall</i>                     | IoT, Aplikasi <i>Mobile</i> | Pengembangan sistem tempat sampah pintar berbasis IoT untuk meningkatkan efisiensi pengumpulan sampah di kampus.                | Sensor ultrasonik, notifikasi otomatis, pengurangan inspeksi manual.                                                                             |

Beberapa penelitian menunjukkan bahwa, meskipun sistem monitoring tempat sampah berbasis IoT dengan integrasi notifikasi *real-time* masih terbatas, pengembangan sistem digital dalam bidang pengelolaan sampah masih didominasi oleh aspek pencatatan data dan transaksi. Hal ini menunjukkan bahwa ada peluang untuk membangun sistem baru yang lebih berfokus pada pemantauan langsung kapasitas tempat sampah dan otomatisasi pemberitahuan kepada petugas kebersihan.

Akibatnya, penelitian ini mengembangkan aplikasi tempat sampah cerdas berbasis ponsel yang terhubung dengan sensor IoT untuk mendeteksi kapasitas tempat sampah secara *real-time* dan mengirimkan notifikasi otomatis kepada petugas kebersihan di lingkungan Fakultas Teknologi Industri Universitas Islam Indonesia (FTI UII). Melalui pengurangan inspeksi manual dan peningkatan kecepatan respons petugas terhadap kondisi tempat sampah, sistem ini diharapkan dapat meningkatkan efisiensi pengelolaan kebersihan kampus.

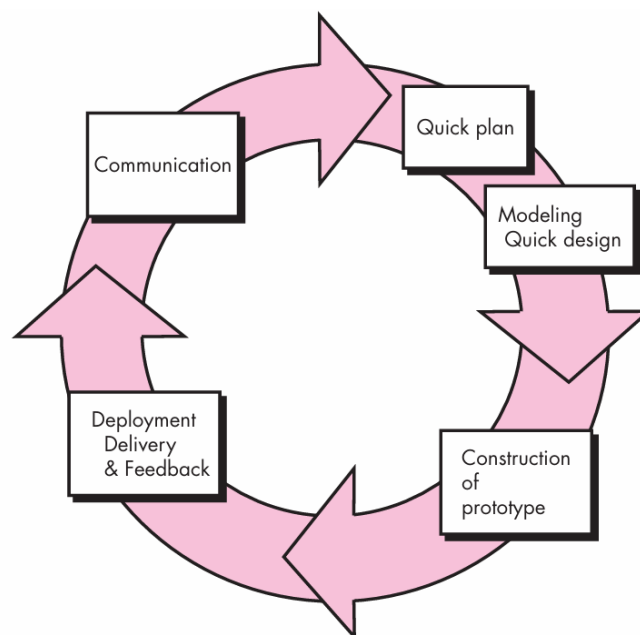
Dengan memanfaatkan kombinasi teknologi IoT dan aplikasi *mobile* Android, penelitian ini mampu melakukan pemantauan kapasitas tempat sampah secara *real-time*, yang belum banyak diterapkan pada penelitian sebelumnya. Penelitian ini juga berfokus pada peningkatan produktivitas petugas kebersihan melalui penyediaan informasi kondisi tempat sampah secara cepat dan akurat, sehingga mendukung pengelolaan kebersihan kampus secara lebih optimal.

### BAB III

## METODOLOGI

Pengembangan sebuah aplikasi memerlukan sebuah metodologi agar proses yang direncanakan berjalan dengan tepat dan lancar. Berkaitan dengan pengembangan aplikasi berbasis *mobile*, penelitian ini dilakukan dengan menggunakan metode *prototyping*. Metode *prototyping* merupakan pendekatan dalam pengembangan perangkat lunak yang melibatkan pembuatan model awal dari sistem (Alda et al., 2023). Dipilihnya metode ini karena *prototyping* dapat dilakukan secara iteratif, artinya pengguna dilibatkan secara langsung dalam proses evaluasi dan penyempurnaan desain antarmuka maupun fungsionalitas sistem.

Dengan kata lain, ketika model rancangan dibuat untuk pertama kalinya, fokusnya adalah menguji konsep desain, kemudian apabila model desain tidak sesuai dengan harapan pengguna, pengujian terus dilakukan sampai konsep desain secara keseluruhan dapat diterima dengan baik dan puas bagi pengguna. Menurut buku Roger S. Pressman dengan judul “*Software Engineering A Practitioner’s Approach Seventh Edition*” metode *prototyping* terdiri dari lima tahapan utama: *communication* (komunikasi), *quick plan* (perencanaan cepat), *modeling quick design* (desain cepat), *construction of prototype* (pembangunan prototipe), *deployment delivery & feedback* (penerapan, dan umpan balik). Setiap tahap dilakukan secara bertahap dan berulang, sehingga hasil dari satu tahap dapat dievaluasi dan diperbaiki pada tahap berikutnya.



Gambar 3.1 Alur *Prototyping*

Sumber: (Pressman, 2010)

### 3.1 Communication (Komunikasi)

Sebagai dasar perancangan sistem, tahap komunikasi bertujuan untuk mengidentifikasi kebutuhan pengguna dan masalah yang terjadi di lapangan. Pada tahap komunikasi, peneliti terlibat langsung dengan pengguna sistem melalui observasi lapangan dan wawancara semi-terstruktur. Kegiatan ini dilakukan untuk memahami alur kerja petugas kebersihan, kendala yang dihadapi, serta kebutuhan sistem yang diharapkan.

#### a. Observasi

Observasi dilakukan secara langsung di area Fakultas Teknologi Industri Universitas Islam Indonesia selama dua hari. Kegiatan observasi dilakukan dengan memperhatikan secara langsung aktivitas petugas kebersihan dalam menjalankan tugas sehari-hari. Fokus observasi meliputi pengecekan tempat sampah, frekuensi pengangkutan sampah, kondisi tempat sampah ketika penuh, dan lokasi tempat sampah di area gedung. Selama observasi, peneliti mencatat bahwa pengecekan kondisi tempat sampah masih dilakukan secara manual dengan cara petugas berkeliling ke setiap lokasi.

Berdasarkan hasil observasi tersebut, diketahui bahwa proses pemeriksaan kondisi tempat sampah masih dilakukan secara manual yaitu dengan cara petugas berkeliling untuk memeriksa setiap tempat sampah. Proses ini membutuhkan waktu dan tenaga yang lebih besar. Selain itu, tidak adanya informasi secara *real-time* menyebabkan petugas kesulitan menentukan prioritas penanganan, sehingga berpotensi menimbulkan penumpukan sampah di beberapa titik.

#### b. Wawancara

Setelah melakukan observasi, dilanjutkan dengan melakukan wawancara semi-terstruktur dengan petugas kebersihan yang merupakan pengguna utama sistem. Tujuannya untuk mendapatkan informasi tentang kebutuhan sistem dan kendala yang dihadapi. Selain itu, wawancara juga bertujuan untuk melengkapi data hasil observasi dan memastikan masalah utama yang ada di lapangan serta mendapatkan masukan awal untuk pengembangan fitur aplikasi.

Wawancara dilakukan kepada sembilan responden yang terdiri dari lima orang petugas kebersihan serta empat mahasiswa yang berasal dari dua program studi, yaitu dua mahasiswa Teknik Elektro dan dua mahasiswa Informatika dengan tatap muka secara langsung di lingkungan kampus pada jam istirahat. Pemilihan responden dilakukan secara *purposive* dengan mempertimbangkan keterlibatan langsung terhadap penggunaan atau interaksi dengan fasilitas tempat sampah di area kampus.

Petugas kebersihan dipilih dengan kriteria: (1) aktif bekerja di lingkungan Fakultas Teknologi Industri, (2) terlibat langsung dalam pengelolaan dan pengangkutan sampah, serta (3) memiliki pengalaman kerja minimal 1 tahun. Kriteria ini digunakan karena petugas kebersihan merupakan pengguna utama sistem yang memahami kondisi operasional di lapangan.

Sementara itu, mahasiswa dipilih dari dua program studi, yaitu Teknik Elektro dan Informatika. Mahasiswa Teknik Elektro dipilih karena memiliki pemahaman terhadap perangkat keras dan teknologi *Internet of Things* (IoT), sehingga dapat memberikan pandangan terkait aspek teknis pada alat tempat sampah cerdas. Sedangkan mahasiswa Informatika dipilih karena memiliki pemahaman dalam pengembangan sistem dan aplikasi, sehingga dapat memberikan masukan terkait antarmuka dan fungsionalitas aplikasi *mobile*. Peneliti menyiapkan daftar pertanyaan sebagai panduan wawancara, namun tetap memberikan kesempatan kepada responden untuk menyampaikan pengalaman dan pendapat secara lebih bebas. Setiap sesi wawancara berlangsung selama kurang lebih 15–20 menit. Selama proses wawancara, peneliti mencatat jawaban responden untuk kemudian dianalisis. Hasil wawancara selanjutnya dikelompokkan ke dalam beberapa tema utama, seperti kendala dalam pengelolaan sampah, kebutuhan akan informasi kondisi tempat sampah secara *real-time*, serta harapan pengguna terhadap fitur dan kemudahan penggunaan aplikasi *mobile*.

Daftar pertanyaan wawancara ditunjukkan pada table berikut:

Tabel 3.1 Daftar Pertanyaan Wawancara

| No | Pertanyaan                                                                                                                                  |
|----|---------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Bagaimana pendapat Bapak/Ibu mengenai kondisi pengelolaan sampah di lingkungan kampus saat ini?                                             |
| 2  | Masalah apa saja yang paling sering Bapak/Ibu hadapi terkait pengelolaan tempat sampah?                                                     |
| 3  | Apakah Bapak/Ibu pernah mengalami kesulitan ketika tempat sampah dalam kondisi penuh atau rusak? Jika ya, bagaimana biasanya penanganannya? |
| 4  | Menurut Bapak/Ibu, aspek apa saja yang perlu diperbaiki atau ditingkatkan dalam pengelolaan tempat sampah di kampus?                        |
| 5  | Apabila tersedia aplikasi yang dapat memantau kondisi tempat sampah, fitur apa saja yang menurut Bapak/Ibu perlu disediakan?                |
| 6  | Bagaimana pendapat Bapak/Ibu mengenai fitur pelaporan kondisi tempat sampah, seperti penuh atau rusak, melalui aplikasi <i>mobile</i> ?     |

|    |                                                                                                                                                       |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7  | Apakah Bapak/Ibu pernah menggunakan aplikasi untuk mendukung pekerjaan sehari-hari? Jika pernah, aplikasi apa dan bagaimana pengalaman penggunaannya? |
| 8  | Menurut Bapak/Ibu, seperti apa tampilan aplikasi yang mudah digunakan untuk mendukung pekerjaan di lapangan?                                          |
| 9  | Menurut Bapak/Ibu, apa satu hal terpenting yang harus dimiliki oleh aplikasi tempat sampah cerdas?                                                    |
| 10 | Apakah terdapat saran atau ide fitur lain yang dapat menjadi masukan untuk pengembangan sistem pengelolaan sampah di kampus?                          |

Proses pengumpulan kebutuhan ini selaras dengan prinsip dari metode *prototyping* yang menekankan keterlibatan pengguna dari awal dan secara berkala selama tahap pengembangan sistem (Ramdhani et al., 2022). Pertanyaan wawancara disusun untuk menggali informasi terkait permasalahan pengelolaan tempat sampah, kebutuhan sistem, serta harapan pengguna terhadap aplikasi yang akan dikembangkan.

c. Ringkasan Hasil Wawancara

Tabel 3.2 Hasil Wawancara

| No | Kode responden | Kategori responden | Masalah Utama                                                                                                                                                                              | Kebutuhan Fitur                                                                                                          | Kebutuhan Sistem Berdasarkan Temuan                                                                                                                                                |
|----|----------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | P1             | Petugas kebersihan | Pemilahan sampah pernah diterapkan, tetapi tidak berjalan lagi sehingga sampah kembali tercampur. Saat terdapat kegiatan kampus, volume sampah meningkat dan dapat menyebabkan penumpukan. | Notifikasi ketika tempat sampah penuh, informasi tempat sampah, dan tampilan aplikasi yang mudah dipahami secara visual. | Sistem perlu menyediakan <i>dashboard</i> monitoring, notifikasi otomatis, serta informasi identitas tempat sampah agar petugas dapat mengetahui titik yang perlu segera ditangani |
| 2. | P2             | Petugas kebersihan | Sampah masih tercampur dan pengecekan tempat                                                                                                                                               | Pemantauan kapasitas tempat sampah,                                                                                      | Sistem perlu menampilkan kapasitas tempat                                                                                                                                          |

|    |    |                    |                                                                                                                                                                                                            |                                                                                                                         |                                                                                                                                                              |
|----|----|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |    |                    | sampah masih dilakukan secara manual berdasarkan jadwal. Petugas harus berkeliling untuk mengetahui kondisi tempat sampah.                                                                                 | notifikasi ketika penuh, informasi tempat sampah, dan tampilan sederhana berbasis visual.                               | sampah secara <i>real-time</i> melalui aplikasi agar petugas tidak harus selalu melakukan pengecekan manual ke setiap lokasi. Serta tampilan yang sederhana. |
| 3. | P3 | Petugas kebersihan | Sampah yang dikumpulkan masih sering tercampur, termasuk sampah basah atau cairan. Petugas juga menyampaikan adanya risiko ketika sampah tercampur dengan benda yang dapat membahayakan saat pengangkutan. | Informasi kapasitas, indikator kondisi tempat sampah, pemilahan jenis sampah, dan tampilan visual yang jelas.           | Sistem perlu menampilkan informasi kapasitas, indikator kondisi tempat sampah, pemilahan jenis sampah.                                                       |
| 4. | P4 | Petugas kebersihan | Sampah belum dipilah dengan baik, terutama sampah basah dan kering. Ketika tempat sampah penuh atau perlengkapan sampah tidak tersedia, petugas                                                            | Pemilahan sampah, notifikasi ketika tempat sampah penuh, dan tampilan aplikasi yang mudah dipahami oleh semua pengguna. | Sistem perlu menyediakan informasi kondisi tempat sampah, notifikasi ketika tempat sampah penuh, dan tampilan aplikasi yang mudah                            |

|    |    |                       |                                                                                                                                                                                                                     |                                                                                                                                                                     |                                                                                                                                                                                                       |
|----|----|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |    |                       | perlu mencari plastik tambahan atau meminta bantuan.                                                                                                                                                                |                                                                                                                                                                     | dipahami oleh semua pengguna.                                                                                                                                                                         |
| 5. | P5 | Petugas kebersihan    | Pada area luar, sampah yang sering ditemukan adalah sampah daun. Kendala yang muncul berkaitan dengan keterbatasan wadah dan proses pengangkutan yang bergantung pada kendaraan.                                    | Pemantauan volume sampah, notifikasi ketika penuh, pemantauan <i>real-time</i> , dan informasi jenis sampah.                                                        | Sistem perlu membantu petugas menentukan prioritas pengangkutan melalui informasi kapasitas tempat sampah secara <i>real-time</i> , notifikasi ketika tempat sampah penuh dan informasi jenis sampah. |
| 6. | M1 | Mahasiswa informatika | Mahasiswa masih bingung dalam memilah sampah karena tempat sampah memiliki warna yang sama dan tidak selalu memiliki keterangan yang jelas. Ketika tempat sampah penuh, mahasiswa harus mencari tempat sampah lain. | Informasi pemilahan yang jelas, laporan atau pemberitahuan kondisi tempat sampah, pemantauan <i>real-time</i> , serta tampilan gabungan gambar dan tulisan singkat. | Sistem perlu menampilkan informasi jenis sampah dan kondisi tempat sampah dengan ikon atau keterangan singkat agar mudah dipahami oleh pengguna.                                                      |
| 7. | M2 | Mahasiswa informatika | Tempat sampah di kampus sudah                                                                                                                                                                                       | Informasi pemilahan                                                                                                                                                 | Sistem perlu menyediakan                                                                                                                                                                              |

|    |    |                   |                                                                                                                                                                                                                                               |                                                                        |                                                                                                                                                    |
|----|----|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
|    |    |                   | tersedia, tetapi informasi pemilahan sampah belum selalu jelas. Pengguna masih dapat keliru saat membuang sampah karena kurangnya petunjuk jenis sampah.                                                                                      | sampah belum jelas dan pemantauan kondisi tempat sampah.               | informasi jenis sampah secara jelas dan menampilkan kondisi tempat sampah.                                                                         |
| 8  | M3 | Mahasiswa elektro | Tempat sampah dinilai sudah tersedia di beberapa titik, tetapi sampah masih digabung dalam satu wadah. Penumpukan sampah juga dapat terjadi setelah kegiatan kampus, terutama ketika kegiatan berlangsung pada hari libur petugas kebersihan. | Pemilahan jenis sampah dan pemantauan kondisi tempat sampah.           | Sistem perlu menampilkan informasi jenis sampah secara jelas dan monitoring agar pengguna dapat memahami kondisi tempat sampah dengan lebih mudah. |
| 9. | M4 | Mahasiswa elektro | Data kapasitas tempat sampah perlu terbaca dengan akurat agar informasi pada aplikasi sesuai dengan                                                                                                                                           | Sensor kapasitas dan pengiriman data yang sesuai dengan tempat sampah. | Sistem perlu membaca data kapasitas dari sensor dan menampilkannya pada aplikasi sesuai dengan tempat                                              |

|  |  |  |                        |  |                                                         |
|--|--|--|------------------------|--|---------------------------------------------------------|
|  |  |  | kondisi<br>sebenarnya. |  | sampah yang<br>terdaftar dan hasil<br>pembacaan sensor. |
|--|--|--|------------------------|--|---------------------------------------------------------|

Berdasarkan hasil observasi lapangan dan wawancara dengan petugas kebersihan serta mahasiswa, diperoleh beberapa temuan penting terkait permasalahan pengelolaan sampah di lingkungan Fakultas Teknologi Industri Universitas Islam Indonesia. Temuan ini menunjukkan apa yang terjadi di lapangan dan berfungsi sebagai dasar untuk menentukan apa yang diperlukan untuk sistem yang akan dibangun.

Hasil wawancara menunjukkan bahwa pemantauan tempat sampah di kampus masih dilakukan secara manual. Petugas kebersihan harus melakukan pengecekan langsung di setiap tempat sampah untuk memastikan tingkat kepenuhan dan kerusakan. Proses tersebut dinilai kurang efisien karena membutuhkan waktu dan tenaga, terutama ketika terdapat banyak lokasi yang harus dipantau. Selain itu, ketika tempat sampah penuh, penanganan dapat terlambat karena belum adanya informasi awal mengenai kondisi tempat sampah sebelum dilakukan pengecekan langsung.

Temuan lain menunjukkan bahwa pemilahan sampah belum berjalan secara optimal. Beberapa responden menyampaikan bahwa sampah masih sering tercampur dalam satu wadah, sehingga proses pengelolaan menjadi kurang terarah. Kebutuhan ini disesuaikan dengan kemampuan perangkat IoT yang digunakan dalam penelitian, yaitu pemantauan dan pemilahan tiga jenis sampah utama berupa sampah organik, botol plastik, dan kaleng.

Petugas kebersihan juga menyampaikan bahwa aplikasi yang mampu memantau kondisi tempat sampah secara *real-time* dapat membantu pekerjaan di lapangan. Informasi kapasitas tempat sampah, status kondisi, serta notifikasi otomatis ketika tempat sampah mendekati penuh dinilai penting untuk membantu petugas menentukan prioritas pembersihan atau pengangkutan sampah.

Selain itu, responden mengharapkan aplikasi yang sederhana dan mudah digunakan. Informasi yang ditampilkan perlu dibuat ringkas, jelas, dan mudah dipahami, seperti status kondisi tempat sampah, persentase kapasitas, jenis sampah, serta informasi identitas tempat sampah. Tampilan aplikasi yang sederhana diperlukan agar pengguna dapat memahami informasi dengan cepat saat bekerja di lapangan.

Secara keseluruhan, hasil wawancara menunjukkan bahwa sistem yang dibutuhkan adalah aplikasi *mobile* yang mampu menampilkan kondisi tempat sampah secara cepat, mudah

diakses, dan sesuai dengan kebutuhan pengguna. Temuan ini digunakan sebagai dasar pada tahap perencanaan cepat, khususnya dalam menentukan kebutuhan fungsional dan non-fungsional sistem.

### 3.2 Quick Plan (Perencanaan Cepat)

Tahap perencanaan cepat (*quick plan*) dilakukan setelah proses komunikasi selesai. Tahap perencanaan cepat dilakukan dengan menganalisis hasil observasi dan wawancara untuk merumuskan kebutuhan fungsional dan non-fungsional sistem yang akan menjadi dasar perancangan *prototype*.

#### Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dibuat berdasarkan observasi lapangan dan wawancara dengan petugas kebersihan pengguna utama sistem di Fakultas Teknologi Industri Universitas Islam Indonesia. Analisis ini bertujuan untuk merumuskan kebutuhan yang harus dipenuhi agar sistem mampu menyelesaikan permasalahan yang ditemukan pada tahap komunikasi, khususnya terkait pemantauan kondisi tempat sampah yang masih dilakukan secara manual.

Berdasarkan hasil identifikasi kebutuhan tersebut, sistem dirancang untuk mendukung dua jenis tempat sampah, yaitu tipe A yang berfungsi untuk memantau kapasitas secara umum, serta tipe B yang dilengkapi dengan sensor pemilahan untuk mendeteksi kapasitas berdasarkan jenis sampah (organik, plastik, dan kaleng). Perbedaan ini disesuaikan dengan spesifikasi perangkat IoT yang tersedia pada masing-masing tempat sampah.

Pada penelitian ini, analisis kebutuhan dibagi menjadi dua kategori utama, yaitu:

#### a. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang terkait langsung dengan fungsi utama sistem yang harus dimiliki agar petugas kebersihan dapat menggunakannya. Kebutuhan fungsional dibuat untuk menjawab masalah yang ditemukan selama observasi dan wawancara, terutama terkait dengan kesulitan petugas dalam memantau kondisi tempat sampah secara manual. Berdasarkan hasil analisis, kebutuhan fungsional sistem meliputi:

##### 1. Fitur Autentikasi Pengguna

Untuk menjaga keamanan data dan memastikan bahwa informasi sistem hanya dapat diakses oleh pengguna yang berwenang, fitur autentikasi pengguna sistem harus menyediakan fitur *login* untuk membatasi akses aplikasi kepada petugas kebersihan.

##### 2. Pemantauan Kapasitas

Pemantauan kapasitas tempat sampah secara *real-time* berdasarkan data yang dikirimkan oleh sensor IoT. Sistem menampilkan informasi kapasitas dalam bentuk persentase serta status kondisi tempat sampah, sehingga petugas kebersihan dapat mengetahui kondisi tempat sampah tanpa harus melakukan pengecekan langsung ke lokasi.

### 3. Pengiriman Notifikasi

Sistem harus dapat mengirimkan notifikasi otomatis kepada petugas kebersihan ketika kapasitas tempat sampah telah mencapai atau melebihi ambang batas tertentu. Fitur ini dimaksudkan untuk membantu petugas mengetahui kondisi tempat sampah yang perlu segera ditangani agar penanganan sampah dapat dilakukan tepat waktu.

### 4. Pencatatan dan Riwayat

Sistem harus dapat mencatat dan menampilkan riwayat aktivitas pembersihan tempat sampah. Ini dapat digunakan untuk menentukan waktu dan lokasi pembersihan yang telah dilakukan serta membantu dalam evaluasi pengelolaan kebersihan kampus.

### 5. Informasi dan Edukasi Pengelolaan Sampah

Sistem menyediakan fitur informasi atau artikel singkat mengenai pengelolaan sampah, khususnya pemilahan sampah organik, plastik, dan kaleng, sebagai bentuk edukasi pendukung bagi pengguna aplikasi.

### 6. Fitur Tambah Tempat Sampah

Sistem menyediakan fitur registrasi perangkat tempat sampah yang digunakan untuk menambahkan data tempat sampah baru melalui aplikasi *mobile*. Data yang ditambahkan meliputi nama tempat sampah, jenis perangkat, jenis tempat sampah, serta kode perangkat atau *path* sensor yang terhubung dengan *Firebase Realtime Database*. Kode perangkat digunakan untuk menghubungkan data identitas tempat sampah dengan data sensor yang dikirimkan oleh perangkat IoT.

## b. Kebutuhan Non-Fungsional

Untuk memastikan bahwa aplikasi dapat digunakan secara optimal oleh petugas kebersihan, kebutuhan non-fungsional adalah kualitas sistem yang penting yang menentukan kenyamanan, keandalan, dan efektivitas aplikasi di lapangan. Kebutuhan ini tidak terkait langsung dengan fungsi utama sistem. Kebutuhan non-fungsional sistem meliputi:

### 1. Kemudahan

Pengguna

*User Interface* (UI) dirancang dengan prinsip yang memudahkan pengguna, sehingga

petugas kebersihan yang tidak memiliki pengetahuan teknis pun bisa mengoperasikan aplikasi dengan mudah. *Navigasi* dirancang secara sederhana, menggunakan ikon yang mudah dimengerti, serta menyajikan informasi dengan cara yang mudah dipahami secara visual. Hal ini penting karena metode *prototyping* mendorong penilaian antarmuka dari awal proses pengembangan (Ramdhani et al., 2022).

## 2. Kinerja

Aplikasi harus beroperasi dengan baik pada perangkat Android menengah, dengan respon antarmuka yang cepat dan juga berjalan stabil.

## 3. Keamanan

Sistem harus menjamin informasi keamanan pengguna agar tidak diakses secara ilegal. Proses masuk dilakukan autentikasi sederhana (*username* dan *password*), dan informasi pemantauan tidak boleh dapat diakses oleh pihak luar yang tidak memiliki izin.

## 4. Reliabilitas

Aplikasi perlu berfungsi secara stabil tanpa kesalahan ketika dijalankan dalam jangka waktu yang panjang. Sistem cadangan untuk penanganan kesalahan akan diterapkan ketika sensor tidak berhasil mengirim data. Sehingga aplikasi tetap dapat dipakai dengan menunjukkan pesan status sementara.

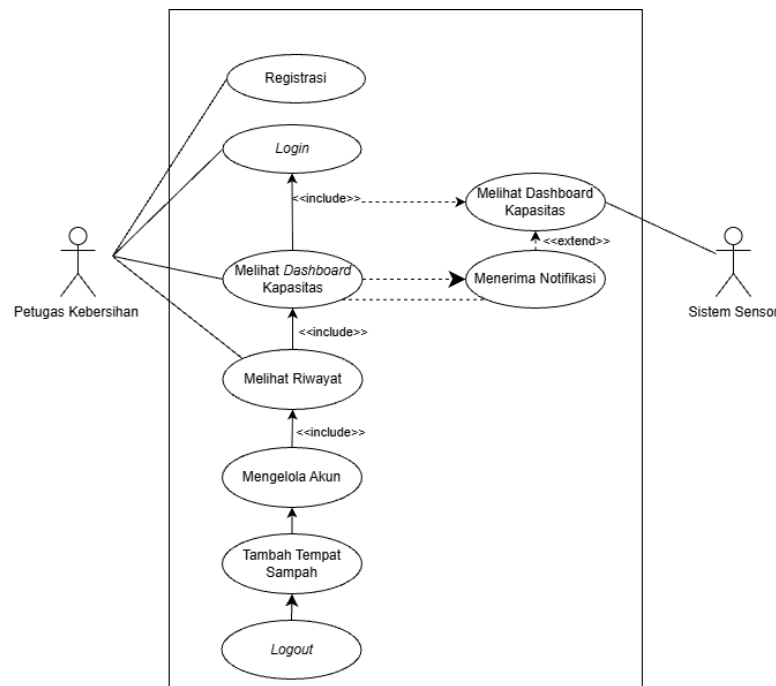
### 3.3 *Modeling Quick Design (Desain Cepat)*

Berdasarkan analisis dan perencanaan yang telah dilakukan sebelumnya, tahap selanjutnya adalah visualisasi atau representasi model sistem yang akan dibuat. Pada tahap *modeling quick design*, rancangan sistem disusun dalam bentuk pemodelan visual agar alur kerja aplikasi, interaksi pengguna, struktur data, dan rancangan antarmuka dapat dipahami secara lebih jelas sebelum masuk ke tahap pembangunan *prototype*. Pemodelan ini menjadi dasar dalam menerjemahkan kebutuhan fungsional dan non-fungsional yang telah diperoleh.

Diagram yang digunakan pada tahap ini meliputi *use case diagram*, *activity diagram*, *Entity Relationship Diagram (ERD)*, dan *wireframe*. *Use case diagram* digunakan untuk menggambarkan aktor yang terlibat serta fitur utama yang dapat diakses oleh pengguna. *Activity diagram* digunakan untuk menjelaskan alur aktivitas pengguna dan sistem pada setiap proses utama aplikasi. *ERD* digunakan untuk menggambarkan struktur data serta hubungan antarentitas yang dibutuhkan sistem. Sementara itu, *wireframe* digunakan sebagai rancangan awal tampilan aplikasi agar susunan halaman dan navigasi dapat terlihat sebelum aplikasi dikembangkan.

Tahap ini penting karena menjadi dasar sebelum aplikasi dikembangkan menggunakan Flutter dan Firebase. Dengan adanya rancangan awal, proses pembangunan *prototype* dapat dilakukan secara lebih terarah. Selain itu, rancangan ini juga membantu memastikan bahwa fitur aplikasi sesuai dengan kebutuhan pengguna, khususnya petugas kebersihan sebagai pengguna utama sistem.

### 3.3.1 Use Case Diagram



Gambar 3.2 Use Case Diagram

*Use case diagram* digunakan untuk menggambarkan hubungan antara aktor dengan fitur-fitur utama yang terdapat pada aplikasi Tempat Sampah Cerdas Berbasis *Mobile*. Pada sistem ini terdapat dua aktor yang terlibat, yaitu petugas kebersihan dan sistem sensor. Petugas kebersihan merupakan aktor utama yang menggunakan aplikasi, sedangkan sistem sensor merupakan aktor pendukung yang berperan dalam mengirimkan data kapasitas tempat sampah ke sistem.

Berdasarkan Gambar 3.2 petugas kebersihan dapat mengakses beberapa fitur utama, yaitu registrasi, *login*, melihat *dashboard* kapasitas, menerima notifikasi, melihat riwayat, mengelola akun, menambahkan tempat sampah, dan *logout*. Fitur registrasi digunakan oleh pengguna baru untuk membuat akun. Setelah akun berhasil dibuat, pengguna dapat melakukan *login* untuk masuk ke dalam aplikasi. Proses *login* diperlukan agar aplikasi hanya dapat diakses oleh pengguna yang terdaftar.

Setelah berhasil *login*, petugas kebersihan dapat melihat *dashboard* kapasitas tempat sampah. *Dashboard* menampilkan informasi kondisi tempat sampah berdasarkan data yang dikirimkan oleh sensor ke *Firebase Realtime Database*. Informasi ini membantu petugas mengetahui kondisi tempat sampah tanpa harus melakukan pengecekan langsung ke setiap lokasi.

Fitur notifikasi digunakan untuk memberikan peringatan kepada petugas kebersihan ketika kapasitas tempat sampah telah mencapai batas tertentu. Dengan adanya notifikasi, petugas dapat mengetahui tempat sampah yang perlu segera ditangani. Fitur riwayat digunakan untuk melihat catatan aktivitas atau kondisi tempat sampah sebelumnya, sehingga pengguna dapat mengetahui data historis dari proses pemantauan.

Fitur tambah tempat sampah digunakan untuk mendaftarkan tempat sampah baru ke dalam aplikasi. Pada fitur ini, pengguna dapat memasukkan data seperti nama tempat sampah, jenis perangkat, jenis tempat sampah, dan kode perangkat atau *path sensor*. Data tersebut diperlukan agar aplikasi dapat membaca data sensor sesuai dengan tempat sampah yang terdaftar. Sementara itu, fitur mengelola akun digunakan untuk melihat dan mengatur informasi akun pengguna, sedangkan fitur *logout* digunakan untuk keluar dari aplikasi.

Sistem sensor berperan dalam mengirimkan data kapasitas tempat sampah ke *database*. Data yang dikirimkan oleh sensor kemudian digunakan oleh aplikasi untuk memperbarui informasi pada *dashboard* dan memicu notifikasi apabila kapasitas telah melewati batas yang ditentukan. Dengan demikian, *use case diagram* pada Gambar 3.2 menunjukkan bahwa aplikasi berfungsi sebagai media penghubung antara petugas kebersihan dan data tempat sampah yang dikirimkan oleh perangkat IoT.

### 3.3.2 Activity Diagram

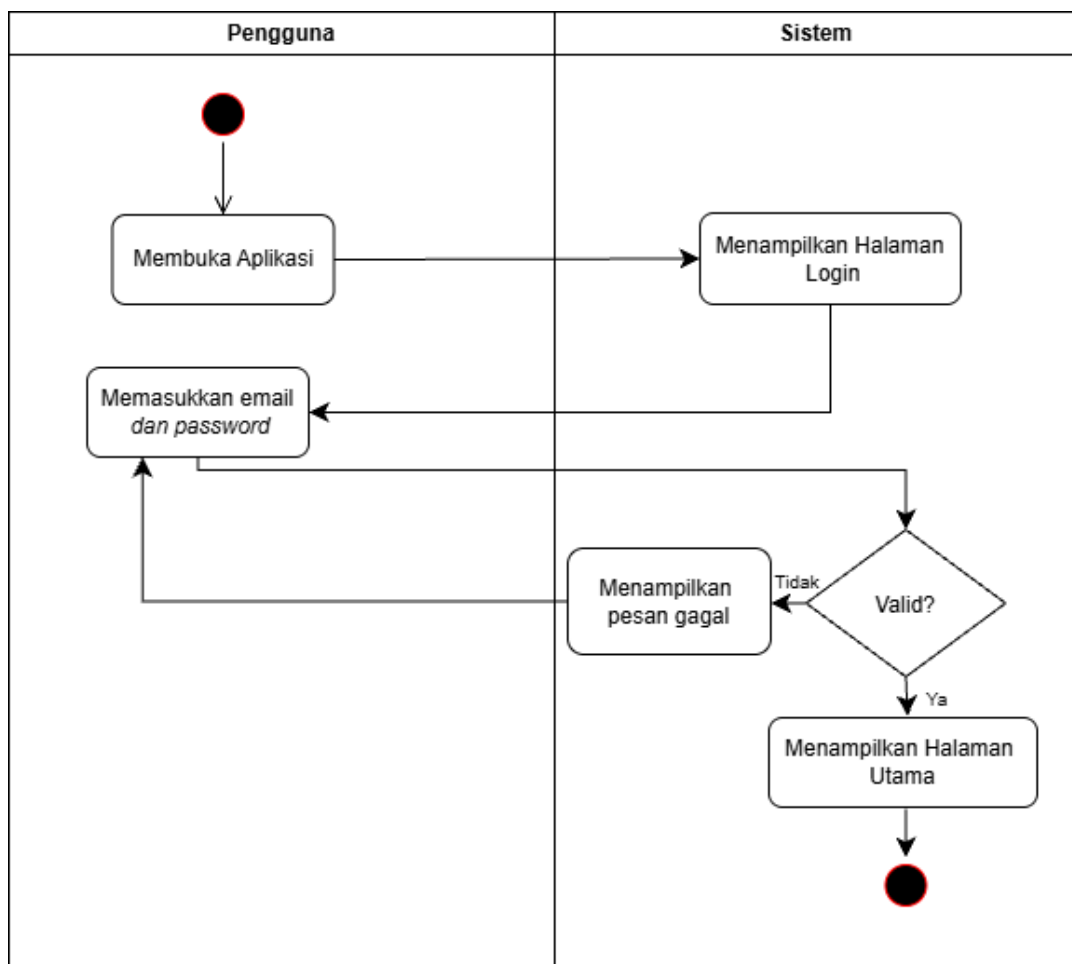
*Activity diagram* digunakan untuk menjelaskan alur proses yang terjadi pada setiap fitur utama aplikasi. Diagram ini menunjukkan aktivitas yang dilakukan oleh pengguna serta respon yang diberikan oleh sistem, sehingga proses kerja aplikasi dapat dipahami secara lebih jelas sebelum masuk ke tahap implementasi.

Pada penelitian ini, *activity diagram* dibuat untuk lima proses utama, yaitu *login*, *dashboard*, notifikasi, riwayat, dan tambah tempat sampah. Pemilihan kelima proses tersebut disesuaikan dengan fitur utama aplikasi yang digunakan oleh petugas kebersihan dalam melakukan pemantauan tempat sampah. Proses *login* menggambarkan alur autentikasi pengguna sebelum masuk ke aplikasi. Proses *dashboard* menggambarkan alur sistem dalam

menampilkan data kapasitas tempat sampah. Proses notifikasi menggambarkan alur pemberian peringatan ketika kapasitas tempat sampah melewati batas tertentu. Proses riwayat menggambarkan alur pengguna dalam melihat catatan aktivitas atau kondisi tempat sampah sebelumnya. Sementara itu, proses tambah tempat sampah menggambarkan alur pengguna dalam mendaftarkan tempat sampah baru agar dapat terhubung dengan data sensor.

Melalui *activity diagram*, setiap alur kerja aplikasi dapat dijelaskan secara berurutan, mulai dari *input* pengguna, proses yang dilakukan oleh sistem, hingga *output* yang ditampilkan kepada pengguna. Diagram ini juga membantu memastikan bahwa setiap fitur memiliki alur yang jelas dan sesuai dengan kebutuhan sistem yang telah ditentukan pada tahap perencanaan cepat.

a. *Activity Diagram Login*



Gambar 3.3 *Activity Diagram Login*

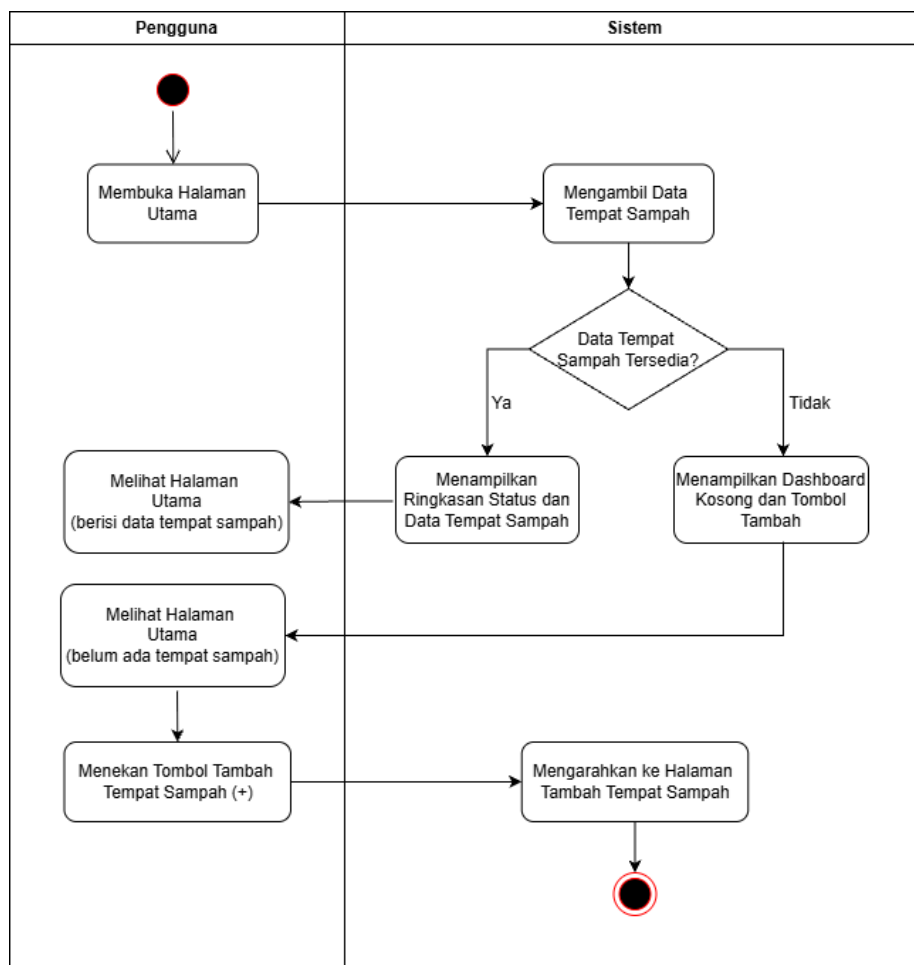
Activity diagram *login* pada Gambar 3.3 menjelaskan alur pengguna ketika akan masuk ke dalam aplikasi. Proses dimulai ketika pengguna membuka aplikasi. Setelah aplikasi

terbuka, sistem menampilkan halaman *login* sebagai halaman awal. Pada halaman ini, pengguna diminta untuk memasukkan email dan *password*.

Setelah email dan *password* diisi, pengguna menekan tombol *login*. Sistem kemudian melakukan validasi terhadap data yang dimasukkan. Validasi dilakukan untuk memastikan bahwa email dan *password* sesuai dengan data akun yang telah terdaftar pada sistem. Apabila data yang dimasukkan benar, sistem akan mengarahkan pengguna ke halaman *dashboard*.

Apabila email atau *password* yang dimasukkan tidak sesuai, sistem akan menampilkan pesan kesalahan. Setelah itu, pengguna dapat mengisi ulang email dan *password* dengan data yang benar. Alur ini menunjukkan bahwa proses *login* berfungsi sebagai pengamanan awal agar aplikasi hanya dapat diakses oleh pengguna yang memiliki akun terdaftar.

#### b. Activity Diagram Dashboard



Gambar 3.4 Activity Diagram Dashboard

Activity diagram dashboard pada Gambar 3.4 menjelaskan alur ketika pengguna berhasil masuk ke halaman utama aplikasi. Dashboard merupakan halaman utama yang digunakan

untuk menampilkan ringkasan kondisi tempat sampah dan menyediakan akses cepat ke fitur penambahan tempat sampah.

Proses dimulai ketika pengguna berhasil *login* ke dalam aplikasi. Setelah itu, sistem menampilkan halaman *dashboard* yang berisi sapaan pengguna, ikon notifikasi, ringkasan status tempat sampah, akses cepat, serta menu navigasi bawah. Pada bagian ringkasan status, sistem menampilkan jumlah tempat sampah penuh, jumlah tempat sampah kosong, dan total tempat sampah yang terdaftar pada aplikasi.

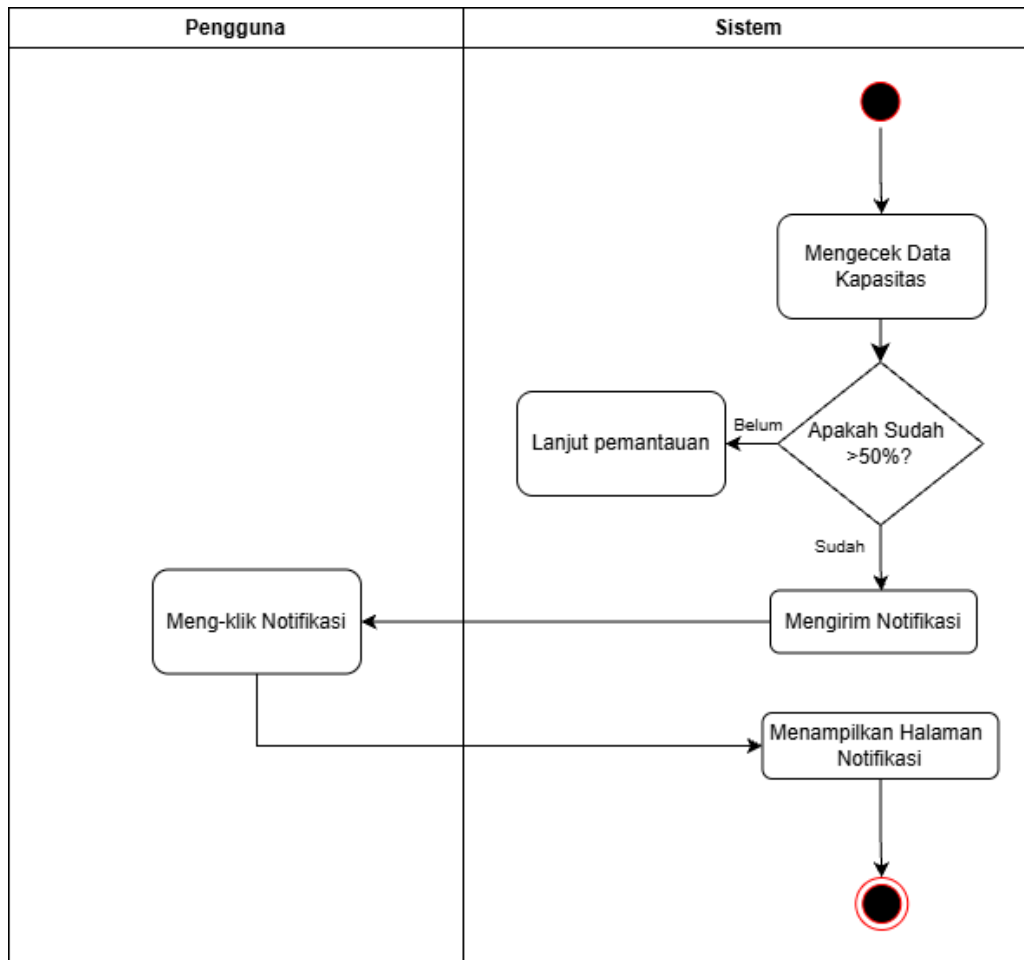
Setelah halaman *dashboard* ditampilkan, sistem mengambil data tempat sampah dari Firebase Realtime Database. Data tersebut digunakan untuk menghitung jumlah tempat sampah berdasarkan status kapasitasnya. Apabila terdapat data tempat sampah yang sudah terdaftar, sistem akan menampilkan informasi kondisi tempat sampah sesuai dengan data yang diterima dari sensor.

Namun, apabila belum terdapat tempat sampah yang terdaftar, sistem akan menampilkan kondisi kosong pada *dashboard*. Kondisi ini ditunjukkan dengan nilai ringkasan status yang masih bernilai 0, pesan “Belum ada Tempat Sampah”, serta tombol “Tambah Tempat Sampah Pertama”. Tombol tersebut digunakan untuk mengarahkan pengguna ke halaman tambah tempat sampah agar pengguna dapat mendaftarkan tempat sampah baru ke dalam sistem.

Selain melalui tombol tersebut, pengguna juga dapat menggunakan ikon tambah (+) pada bagian akses cepat untuk membuka *form* tambah tempat sampah. Setelah tempat sampah berhasil ditambahkan dan terhubung dengan data sensor, sistem akan memperbarui tampilan *dashboard* sesuai dengan data terbaru dari Firebase.

Dengan alur ini, *dashboard* tidak hanya berfungsi untuk menampilkan informasi kapasitas tempat sampah, tetapi juga memberikan informasi awal ketika belum ada data tempat sampah yang terdaftar. Hal ini memudahkan pengguna memahami kondisi sistem dan mengetahui langkah yang harus dilakukan, yaitu menambahkan tempat sampah terlebih dahulu agar data monitoring dapat ditampilkan.

### c. Activity Diagram Notifikasi



Gambar 3.5 Activity Diagram Notifikasi

Activity diagram notifikasi pada Gambar 3.5 menjelaskan proses sistem dalam memberikan peringatan kepada pengguna ketika kapasitas tempat sampah telah mencapai batas tertentu. Proses dimulai ketika sistem membaca data kapasitas tempat sampah dari sensor yang tersimpan pada Firebase Realtime Database. Data kapasitas tersebut kemudian diperiksa oleh sistem untuk mengetahui apakah nilai kapasitas sudah melewati ambang batas notifikasi.

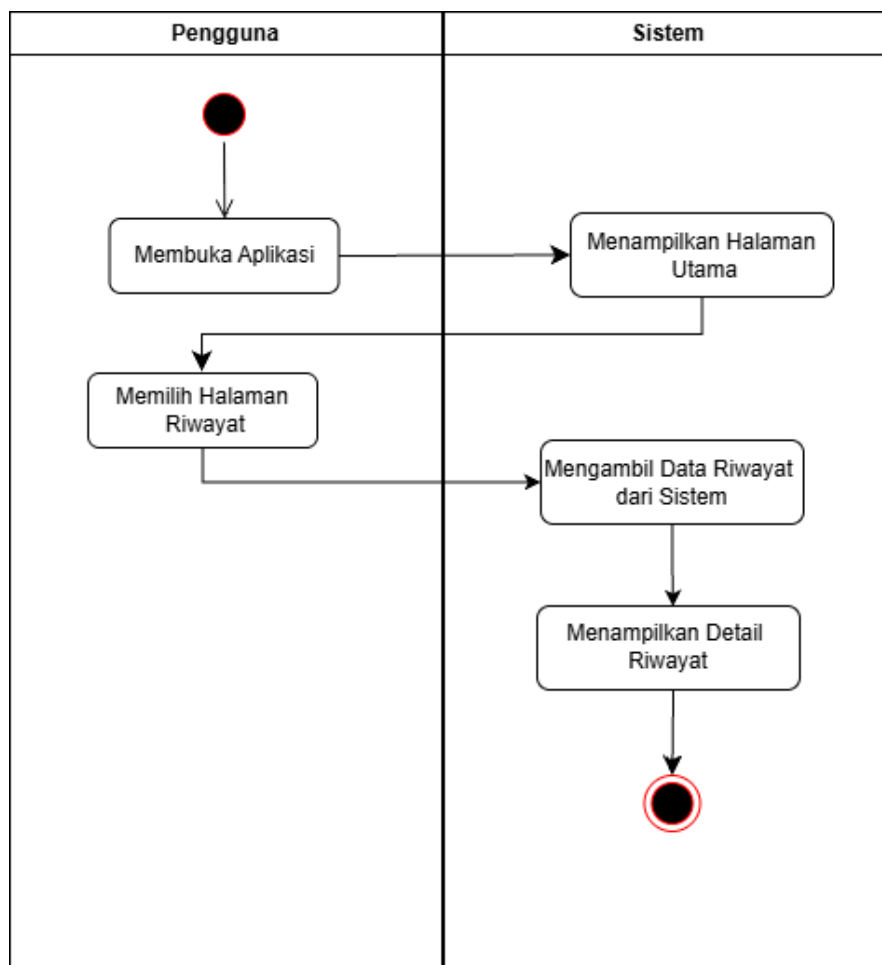
Pada penelitian ini, ambang batas notifikasi ditentukan sebesar 80%. Apabila kapasitas tempat sampah masih di bawah atau sama dengan 80%, sistem tidak mengirimkan notifikasi dan tetap melakukan pemantauan secara berkala. Namun, apabila kapasitas tempat sampah sudah melebihi 80%, sistem akan mengirimkan notifikasi kepada pengguna.

Notifikasi tersebut berisi informasi bahwa tempat sampah penuh atau perlu segera dikosongkan. Dengan adanya notifikasi ini, petugas kebersihan dapat mengetahui tempat

sampah yang membutuhkan penanganan lebih cepat tanpa harus membuka *dashboard* secara terus-menerus atau memeriksa tempat sampah secara manual.

Ambang batas 80% digunakan karena pada kondisi tersebut tempat sampah sudah mendekati penuh, sehingga perlu segera ditangani sebelum benar-benar penuh dan menyebabkan penumpukan sampah. Dengan demikian, fitur notifikasi membantu petugas menentukan prioritas pengangkutan secara lebih cepat dan terarah.

d. *Activity Diagram Riwayat*



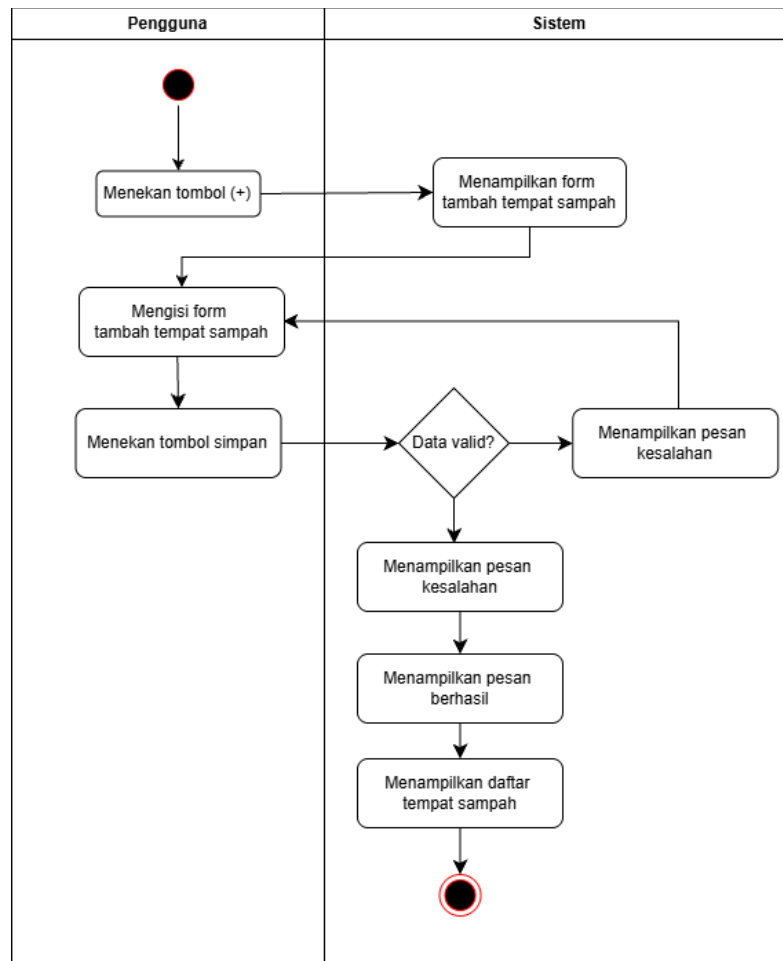
Gambar 3.6 *Activity Diagram Riwayat*

*Activity diagram* riwayat pada Gambar 3.6 menjelaskan alur pengguna ketika ingin melihat catatan aktivitas atau kondisi tempat sampah sebelumnya. Proses dimulai ketika pengguna membuka aplikasi dan memilih menu riwayat.

Setelah menu riwayat dipilih, sistem mengambil data riwayat dari *database*. Data yang ditampilkan dapat berupa waktu aktivitas, nama tempat sampah, status kapasitas, serta kondisi tempat sampah yang tercatat sebelumnya. Setelah data berhasil diambil, sistem menampilkan daftar riwayat kepada pengguna.

Fitur riwayat berfungsi sebagai dokumentasi aktivitas dan kondisi tempat sampah. Melalui fitur ini, pengguna dapat mengetahui data sebelumnya, seperti tempat sampah yang pernah penuh atau aktivitas pembersihan yang telah dilakukan. Data riwayat juga dapat digunakan sebagai bahan evaluasi untuk mengetahui pola kepenuhan tempat sampah di area kampus.

e. *Activity Diagram Tambah Tempat Sampah*



Gambar 3.7 *Activity Diagram Tambah Tempat Sampah*

*Activity diagram* tambah tempat sampah pada Gambar 3.7 menjelaskan proses pengguna dalam menambahkan data tempat sampah baru ke dalam aplikasi. Proses dimulai ketika pengguna menekan tombol tambah atau ikon (+) pada halaman *dashboard*. Setelah itu, sistem menampilkan *form* tambah tempat sampah.

Pada *form* tersebut, pengguna mengisi data yang dibutuhkan, seperti nama tempat sampah, jenis perangkat, jenis tempat sampah, dan kode perangkat atau *path sensor*. *Path sensor* digunakan untuk menghubungkan data tempat sampah pada aplikasi dengan data sensor yang tersimpan pada Firebase Realtime Database. Alamat ini dibutuhkan agar aplikasi

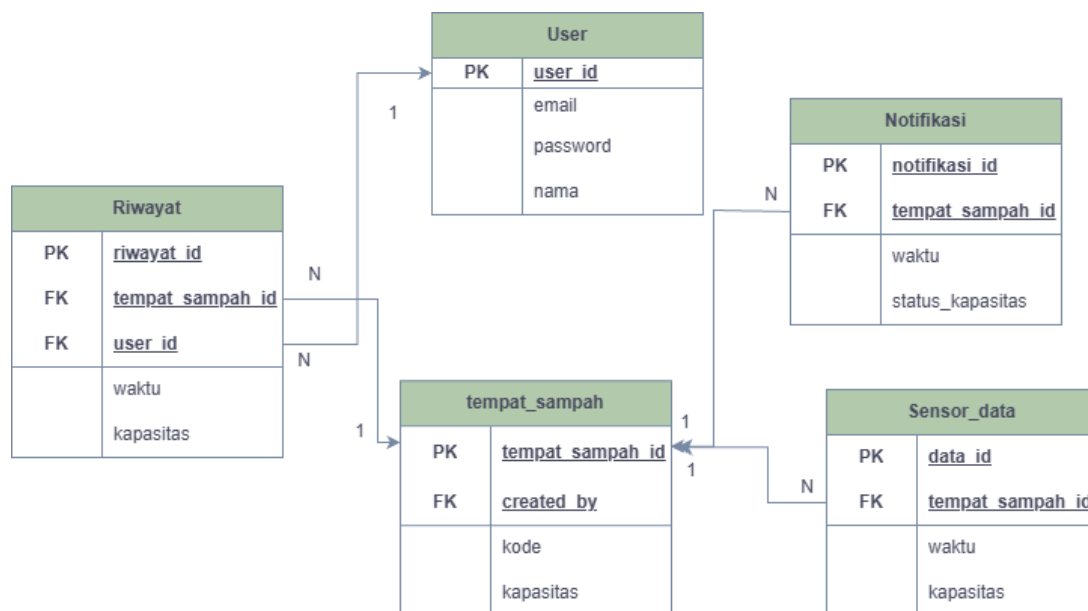
dapat membaca data kapasitas dari tempat sampah yang sesuai dan menampilkannya pada *dashboard*.

Setelah data diisi, pengguna menekan tombol simpan. Sistem kemudian memeriksa kelengkapan data yang dimasukkan. Apabila terdapat data yang belum lengkap atau tidak sesuai, sistem akan menampilkan pesan kesalahan agar pengguna dapat memperbaiki *input*. Apabila data sudah lengkap dan sesuai, sistem menyimpan data tempat sampah ke Firebase Realtime Database.

Setelah data tempat sampah berhasil disimpan, sistem memperbarui daftar tempat sampah pada aplikasi. Tempat sampah yang baru ditambahkan dapat muncul pada *dashboard* sebagai data tempat sampah yang terdaftar. Namun, informasi kapasitas tempat sampah baru dapat ditampilkan apabila perangkat IoT atau sensor sudah mengirimkan data kapasitas ke Firebase Realtime Database.

Fitur ini penting karena aplikasi hanya dapat menampilkan data monitoring dari tempat sampah yang sudah terdaftar. Dengan adanya fitur tambah tempat sampah, pengguna dapat mendaftarkan perangkat baru sehingga data sensor dapat dibaca dan ditampilkan pada dashboard sesuai dengan data yang dikirimkan oleh perangkat IoT.

### 3.3.3 Entity Relationship Diagram (ERD)



Gambar 3.8 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* atau ERD digunakan untuk menggambarkan rancangan struktur data yang digunakan pada aplikasi. ERD menjelaskan entitas apa saja yang terdapat dalam sistem, atribut yang dimiliki setiap entitas, serta hubungan antar entitas. Rancangan ini

dibuat agar struktur data yang digunakan pada aplikasi dapat dipahami dengan lebih jelas sebelum diterapkan ke dalam Firebase Realtime Database. Dengan adanya ERD, proses penyimpanan, pengambilan, dan pengelolaan data dapat dirancang secara lebih terarah.

Meskipun aplikasi ini menggunakan Firebase Realtime Database yang berbasis NoSQL, ERD tetap digunakan sebagai rancangan konseptual. ERD pada penelitian ini tidak dimaksudkan untuk menggambarkan tabel relasional seperti pada MySQL, tetapi digunakan untuk menjelaskan hubungan antar data yang dibutuhkan oleh sistem. Struktur data pada Firebase nantinya dapat disesuaikan ke dalam bentuk node atau JSON, tetapi hubungan antar data tetap perlu dipahami terlebih dahulu. Oleh karena itu, ERD digunakan sebagai alat bantu untuk memperjelas keterkaitan antara data pengguna, tempat sampah, sensor, notifikasi, dan riwayat.

Berdasarkan Gambar 3.8, terdapat lima entitas utama dalam rancangan sistem, yaitu User, Tempat Sampah, Sensor Data, Notifikasi, dan Riwayat. Entitas User digunakan untuk menyimpan data pengguna aplikasi, seperti user id, email, dan nama pengguna. Data ini digunakan untuk mengidentifikasi pengguna yang mengakses aplikasi, baik pada proses registrasi, *login*, maupun pengelolaan akun. Proses autentikasi pengguna dilakukan melalui Firebase Authentication, sehingga data akun pengguna dapat dikelola secara lebih aman.

Entitas Tempat Sampah digunakan untuk menyimpan data tempat sampah yang terdaftar pada aplikasi. Data yang disimpan pada entitas ini meliputi id tempat sampah, user pembuat data, kode atau *path sensor*, serta kapasitas terkini. Entitas ini menjadi pusat dari proses monitoring karena setiap tempat sampah yang ditambahkan akan dihubungkan dengan data sensor yang dikirimkan oleh perangkat IoT. Dengan adanya data tempat sampah, aplikasi dapat mengetahui perangkat mana yang harus dipantau dan ditampilkan pada *dashboard*.

Entitas Sensor Data digunakan untuk menyimpan data kapasitas yang dikirimkan oleh perangkat IoT. Data yang disimpan meliputi id data sensor, id tempat sampah, waktu pembacaan, dan nilai kapasitas. Atribut kapasitas pada entitas ini digunakan untuk mencatat hasil pembacaan sensor pada waktu tertentu. Dengan adanya data sensor, sistem dapat mengetahui perubahan kapasitas tempat sampah dari waktu ke waktu dan menampilkan kondisi terbaru pada aplikasi.

Entitas Notifikasi digunakan untuk menyimpan data peringatan yang muncul ketika kapasitas tempat sampah mencapai batas tertentu. Data pada entitas ini meliputi id notifikasi, id tempat sampah, waktu notifikasi, dan status kapasitas. Entitas ini mendukung fitur notifikasi otomatis agar petugas kebersihan dapat mengetahui tempat sampah yang perlu segera

ditangani. Notifikasi akan berkaitan dengan tempat sampah tertentu, sehingga pengguna dapat mengetahui sumber peringatan yang muncul pada aplikasi.

Entitas Riwayat digunakan untuk menyimpan catatan aktivitas atau kondisi tempat sampah sebelumnya. Data yang disimpan pada entitas ini meliputi id riwayat, id tempat sampah, user id, waktu, dan kapasitas. Entitas riwayat berfungsi sebagai dokumentasi data yang pernah tercatat pada sistem, misalnya kondisi kapasitas sebelum tempat sampah dikosongkan atau aktivitas pemantauan yang telah dilakukan. Dengan adanya riwayat, pengguna dapat melihat kembali data sebelumnya sebagai bahan evaluasi pengelolaan tempat sampah.

Relasi antara User dan Tempat Sampah menunjukkan bahwa satu pengguna dapat menambahkan banyak data tempat sampah. Relasi ini bersifat *one-to-many* karena satu user dapat mendaftarkan lebih dari satu tempat sampah ke dalam aplikasi. Sebaliknya, satu data tempat sampah hanya memiliki satu user pembuat data. Relasi ini diperlukan agar sistem dapat mengetahui pengguna yang menambahkan data tempat sampah tertentu.

Relasi antara Tempat Sampah dan Sensor Data menunjukkan bahwa satu tempat sampah dapat memiliki banyak data sensor. Hal ini terjadi karena sensor dapat mengirimkan data kapasitas secara berkala sesuai pembacaan perangkat IoT. Setiap data sensor tetap terhubung dengan satu tempat sampah tertentu agar data kapasitas yang ditampilkan pada dashboard tidak tertukar. Relasi ini mendukung kebutuhan sistem dalam menampilkan kondisi tempat sampah secara *real-time*.

Relasi antara Tempat Sampah dan Notifikasi menunjukkan bahwa satu tempat sampah dapat menghasilkan banyak notifikasi. Notifikasi dapat muncul lebih dari satu kali apabila kapasitas tempat sampah melewati ambang batas pada waktu yang berbeda. Dengan relasi ini, setiap notifikasi yang muncul dapat diketahui asal tempat sampahnya. Hal ini membantu petugas kebersihan menentukan tempat sampah yang perlu segera ditangani.

Relasi antara Tempat Sampah dan Riwayat menunjukkan bahwa satu tempat sampah dapat memiliki banyak catatan riwayat. Riwayat tersebut digunakan untuk menyimpan informasi kondisi atau aktivitas yang pernah terjadi pada tempat sampah tersebut. Selain itu, relasi antara User dan Riwayat menunjukkan bahwa satu pengguna dapat memiliki banyak aktivitas yang tercatat dalam sistem. Dengan adanya relasi ini, data riwayat dapat ditelusuri berdasarkan pengguna dan tempat sampah yang berkaitan.

Secara keseluruhan, ERD pada Gambar 3.8 menunjukkan bahwa data utama dalam aplikasi berpusat pada entitas Tempat Sampah. Entitas tersebut terhubung dengan Sensor Data, Notifikasi, dan Riwayat karena seluruh proses monitoring bergantung pada data kondisi tempat

sampah. Sementara itu, entitas User digunakan untuk mengidentifikasi pengguna yang mengakses aplikasi dan melakukan aktivitas pada sistem. Rancangan ERD ini mendukung kebutuhan aplikasi dalam menampilkan kapasitas tempat sampah, memberikan notifikasi, menyimpan riwayat, dan menghubungkan aplikasi dengan perangkat IoT.

### 3.3.4 Desain Wireframe

*Wireframe* merupakan rancangan awal antarmuka pengguna (*User Interface*) yang digunakan sebagai acuan dalam proses pembuatan *prototype* aplikasi. *Wireframe* berfungsi untuk menggambarkan struktur halaman, tata letak komponen, serta alur navigasi aplikasi tanpa menampilkan detail visual seperti warna dan desain grafis. Pada tahap ini, *wireframe* dibuat untuk memastikan bahwa tampilan aplikasi dapat mendukung kebutuhan pengguna, khususnya petugas kebersihan sebagai pengguna utama sistem.

*Wireframe* pada penelitian ini terdiri dari beberapa halaman utama, yaitu halaman *login* dan registrasi, *dashboard*, notifikasi, pengaturan akun, dan riwayat aktivitas. Setiap halaman dirancang menggunakan elemen dasar seperti teks, ikon, tombol, dan kolom input untuk merepresentasikan fungsi utama sistem secara sederhana dan mudah dipahami oleh pengguna. Rancangan ini menjadi dasar dalam implementasi antarmuka aplikasi menggunakan Flutter pada tahap pembangunan *prototype*.

#### a. *Wireframe* halaman *login* dan *registrasi*



### Gambar 3.9 Wireframe Login dan Registrasi

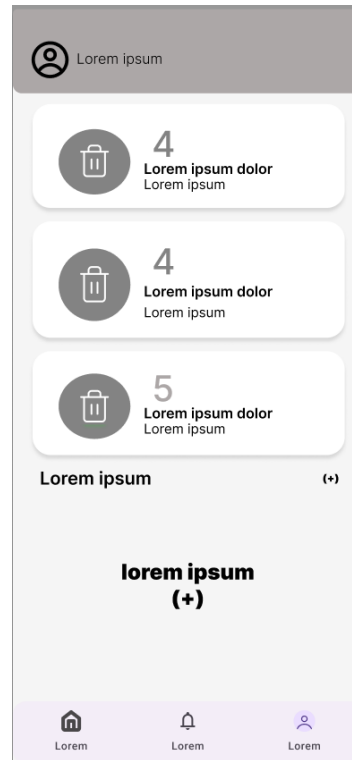
Wireframe halaman *login* dan registrasi pada Gambar 3.9 menunjukkan rancangan awal halaman autentikasi pengguna. Halaman *login* digunakan oleh pengguna yang sudah memiliki akun untuk masuk ke dalam aplikasi, sedangkan halaman registrasi digunakan oleh pengguna baru untuk membuat akun. Kedua halaman ini menjadi gerbang awal sebelum pengguna dapat mengakses fitur utama aplikasi.

Pada halaman *login* terdapat komponen utama berupa kolom email, kolom *password*, dan tombol masuk. Kolom email digunakan untuk memasukkan alamat email pengguna, sedangkan kolom *password* digunakan untuk memasukkan kata sandi. Setelah data diisi, pengguna dapat menekan tombol masuk untuk melanjutkan proses autentikasi. Jika data yang dimasukkan sesuai, pengguna akan diarahkan ke halaman dashboard.

Pada halaman registrasi terdapat beberapa kolom tambahan, seperti nama pengguna, email, nomor telepon, *password*, dan konfirmasi *password*. Kolom tersebut digunakan untuk membuat data akun pengguna baru sebelum pengguna dapat masuk ke aplikasi. Rancangan halaman registrasi dibuat sederhana agar pengguna dapat memahami data yang harus diisi tanpa mengalami kesulitan.

Halaman *login* dan registrasi dirancang dengan tampilan yang ringkas karena fungsi utamanya adalah proses autentikasi. Dengan rancangan ini, pengguna dapat melakukan pendaftaran dan *login* secara lebih mudah sebelum menggunakan fitur monitoring tempat sampah.

#### b. Wireframe halaman *dashboard*



Gambar 3.10 Wireframe Dashboard

*Wireframe* halaman *dashboard* pada Gambar 3.10 dirancang sebagai halaman utama setelah pengguna berhasil masuk ke aplikasi. *Dashboard* berfungsi untuk menampilkan ringkasan kondisi tempat sampah yang terdaftar pada sistem. Halaman ini menjadi pusat informasi awal karena pengguna dapat langsung melihat kondisi umum tempat sampah tanpa harus membuka banyak halaman.

Pada rancangan awal *dashboard*, bagian atas halaman memuat sapaan pengguna dan ikon notifikasi. Sapaan pengguna digunakan untuk menunjukkan bahwa pengguna telah berhasil masuk ke aplikasi, sedangkan ikon notifikasi digunakan sebagai akses menuju halaman pemberitahuan. Di bawah bagian tersebut terdapat ringkasan status yang menampilkan jumlah tempat sampah penuh, jumlah tempat sampah kosong, dan total tempat sampah yang terdaftar.

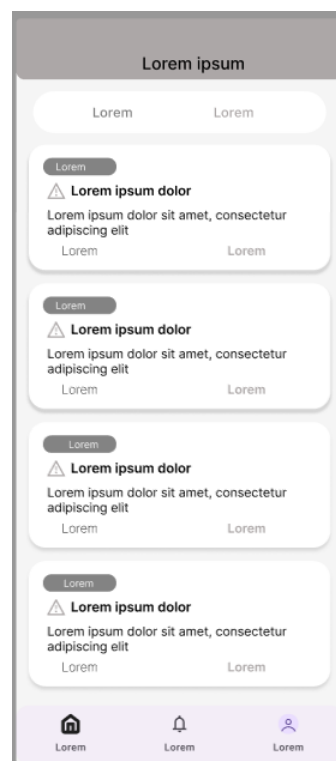
*Dashboard* juga dirancang untuk menyediakan akses cepat menuju fitur tambah tempat sampah. Akses ini dapat berupa tombol tambah atau ikon (+) yang digunakan untuk mendaftarkan tempat sampah baru ke dalam aplikasi. Fitur ini penting karena data monitoring baru dapat ditampilkan apabila tempat sampah sudah terdaftar dan terhubung dengan data sensor.

Pada kondisi awal, apabila belum terdapat data tempat sampah yang terdaftar, *dashboard* dirancang untuk menampilkan kondisi kosong. Kondisi tersebut dapat ditunjukkan melalui

ringkasan status bernilai 0, pesan bahwa belum ada tempat sampah, serta tombol untuk menambahkan tempat sampah pertama. Setelah tempat sampah berhasil ditambahkan dan data sensor tersedia, *dashboard* dapat menampilkan informasi kondisi tempat sampah sesuai data yang diterima oleh sistem.

Rancangan *dashboard* dibuat agar pengguna dapat mengetahui kondisi tempat sampah secara cepat dan mudah. Dengan adanya ringkasan status dan akses tambah tempat sampah, petugas kebersihan dapat memahami apakah sistem sudah memiliki data monitoring atau masih perlu menambahkan tempat sampah terlebih dahulu.

c. *Wireframe* halaman notifikasi



Gambar 3.11 *Wireframe Notifikasi*

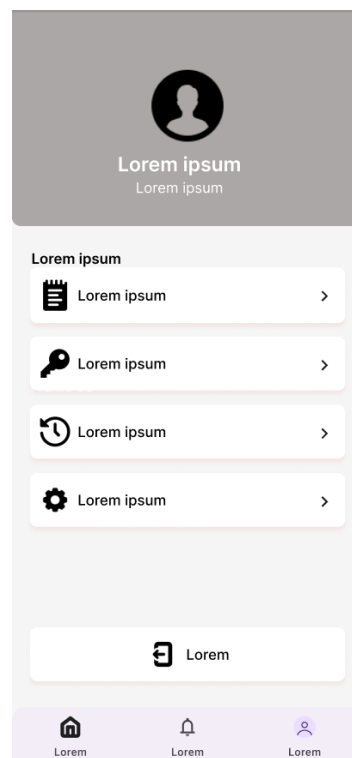
*Wireframe* halaman notifikasi pada Gambar 3.11 dirancang untuk menampilkan pemberitahuan dari sistem kepada pengguna. Halaman ini digunakan ketika terdapat kondisi tempat sampah yang perlu segera diperhatikan, misalnya ketika kapasitas tempat sampah telah melebihi ambang batas 80%. Dengan adanya halaman ini, pengguna dapat mengetahui peringatan tanpa harus memeriksa *dashboard* secara terus-menerus.

Pada rancangan awal halaman notifikasi, informasi yang ditampilkan meliputi nama atau lokasi tempat sampah, status kapasitas, dan waktu munculnya notifikasi. Informasi tersebut disusun secara ringkas agar pengguna dapat langsung memahami tempat sampah

mana yang membutuhkan penanganan. Rancangan ini dibuat untuk membantu pengguna menentukan prioritas pembersihan secara lebih cepat.

Halaman notifikasi dirancang sederhana agar petugas kebersihan dapat membaca informasi peringatan dengan mudah. Ketika terdapat tempat sampah yang kapasitasnya sudah melebihi 80%, sistem akan menampilkan pemberitahuan pada halaman ini. Dengan demikian, halaman notifikasi mendukung tujuan aplikasi dalam membantu proses monitoring tempat sampah secara lebih efisien.

d. *Wireframe* halaman pengaturan



Gambar 3.12 *Wireframe* Pengaturan

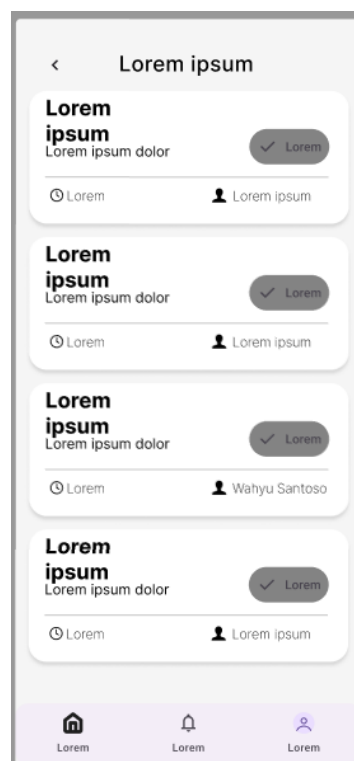
*Wireframe* halaman pengaturan pada Gambar 3.12 dirancang untuk menampilkan dan mengelola informasi akun pengguna. Halaman ini berfungsi sebagai pusat pengaturan akun pada aplikasi. Melalui halaman ini, pengguna dapat melihat informasi profil dan mengakses menu yang berkaitan dengan pengelolaan akun.

Pada rancangan awal halaman pengaturan, komponen yang disediakan meliputi foto profil, nama pengguna, menu edit profil, menu ubah kata sandi, menu riwayat, menu pengaturan tambahan, dan tombol *logout*. Menu edit profil digunakan apabila pengguna ingin memperbarui data akun. Menu ubah kata sandi digunakan untuk menjaga keamanan akun. Menu riwayat digunakan untuk mengarahkan pengguna ke halaman riwayat aktivitas

pembersihan atau catatan kondisi tempat sampah sebelumnya. Sementara tombol *logout* digunakan untuk keluar dari aplikasi setelah selesai digunakan.

Rancangan halaman pengaturan dibuat dengan susunan menu yang sederhana agar pengguna dapat mengakses fitur akun dan riwayat tanpa kesulitan. Hal ini penting karena pengguna utama aplikasi adalah petugas kebersihan yang membutuhkan tampilan praktis dan mudah dipahami. Dengan adanya halaman pengaturan, aplikasi tidak hanya menyediakan fitur monitoring, tetapi juga menyediakan akses untuk mengelola akun dan melihat kembali aktivitas yang pernah tercatat pada sistem.

e. *Wireframe* halaman riwayat



Gambar 3.13 *Wireframe Riwayat*

*Wireframe* halaman riwayat pada Gambar 3.13 dirancang untuk menampilkan catatan aktivitas atau kondisi tempat sampah sebelumnya. Halaman ini berfungsi sebagai dokumentasi data yang pernah tercatat pada sistem. Dengan adanya halaman riwayat, pengguna dapat melihat kembali informasi terkait kondisi tempat sampah dari waktu ke waktu.

Pada rancangan awal halaman riwayat, informasi yang ditampilkan meliputi waktu aktivitas, nama atau lokasi tempat sampah, status kapasitas, dan kondisi terakhir tempat sampah. Data tersebut membantu pengguna mengetahui aktivitas pembersihan atau

pemantauan yang telah dilakukan sebelumnya. Informasi ini juga dapat digunakan untuk melihat pola kepenruhan tempat sampah di area tertentu.

Halaman riwayat dirancang agar pengguna dapat melakukan evaluasi terhadap aktivitas pengelolaan tempat sampah. Apabila terdapat tempat sampah yang sering penuh, data riwayat dapat menjadi dasar untuk menentukan prioritas pembersihan atau kebutuhan penambahan tempat sampah. Dengan demikian, halaman riwayat tidak hanya berfungsi sebagai catatan, tetapi juga sebagai pendukung evaluasi pengelolaan kebersihan.

Berdasarkan rancangan *wireframe* yang telah dibuat, setiap halaman memiliki fungsi yang saling mendukung dalam aplikasi Tempat Sampah Cerdas Berbasis *Mobile*. Halaman *login* dan registrasi digunakan sebagai proses autentikasi pengguna. Halaman dashboard digunakan sebagai pusat informasi kondisi tempat sampah. Halaman notifikasi digunakan untuk menampilkan peringatan dari sistem. Halaman pengaturan digunakan untuk mengelola akun pengguna. Sementara itu, halaman riwayat digunakan untuk menampilkan catatan aktivitas atau kondisi tempat sampah sebelumnya.

*Wireframe* ini menjadi acuan awal dalam pembangunan *prototype* aplikasi. Rancangan yang dibuat pada tahap ini masih dapat mengalami perubahan setelah *prototype* diuji dan pengguna memberikan masukan. Dengan demikian, *wireframe* berperan sebagai dasar perancangan antarmuka sebelum aplikasi diimplementasikan dan dievaluasi pada tahap berikutnya.

### **3.4 Construction of Prototype (Pembangunan Prototipe)**

Hasil dari perancangan *wireframe* yang telah dibuat sebelumnya diterapkan ke dalam bentuk aplikasi yang dapat digunakan dan diuji pada tahap pembuatan *prototype*. Menurut Pressman, (2010), fase ini merupakan bagian penting dari metode *prototype*, di mana model yang telah dirancang diuji secara langsung oleh pengguna untuk mendapatkan umpan balik tentang fitur dan tampilan sistem. Tujuan dari fase ini adalah untuk menghasilkan versi awal aplikasi, atau *prototype*, yang dapat digunakan sebagai alat untuk mengevaluasi kesesuaian sistem dengan kebutuhan pengguna.

Pada penelitian ini, *framework* Flutter digunakan untuk membuat antarmuka dan fungsionalitas aplikasi *mobile*, serta Firebase sebagai layanan penyimpanan data dan autentikasi pengguna. Peneliti Alda et al. (2023) juga menggunakan metode pembuatan *prototype* untuk membuat aplikasi Android berbasis Firebase, di mana hasil dari rancangan

awal diterapkan ke dalam *prototype* fungsional untuk diuji oleh pengguna. Proses pembuatan prototipe meliputi beberapa langkah:

a. Implementasi Struktur Halaman Aplikasi

Halaman utama aplikasi Flutter, seperti registrasi, *login*, *dashboard*, notifikasi, pengaturan akun, dan riwayat aktivitas, dibangun dengan menggunakan komponen dasar seperti *TextField*, *Button*, *ListView*, dan *Card*, yang dirancang dalam tahap *wireframe*. Untuk membuatnya mudah digunakan oleh petugas kebersihan, implementasi dilakukan secara bertahap dengan memperhatikan konsistensi alur navigasi antar halaman. Metode ini sejalan dengan penelitian Alda et al. (2023) yang menemukan bahwa hasil desain digunakan untuk membuat aplikasi interaktif berbasis Android.

b. Integrasi Sensor IoT dengan Firebase

Setelah struktur tampilan selesai, aplikasi dapat terhubung ke Firebase Realtime Database untuk menyimpan dan menampilkan data kapasitas tempat sampah secara *real-time*. Data ini dikirim secara langsung ke aplikasi melalui sensor IoT yang mendeteksi volume sampah di setiap wadah. Saat sensor membaca nilai tertentu, data secara otomatis diperbarui di aplikasi *mobile* dan disimpan di Firebase.

Dengan mekanisme ini, pengguna dapat memantau kondisi tempat sampah tanpa pergi ke lokasi secara manual. Peneliti Ramdhani et al. (2022) menggunakan pendekatan integrasi serupa untuk membangun sistem bank sampah berbasis *prototype* yang menampilkan data transaksi secara langsung dari basis data.

c. Implementasi Fitur Utama Aplikasi

Beberapa fitur utama diimplementasikan, seperti:

- 1) Registrasi dan *Login* untuk proses autentikasi pengguna menggunakan *Firebase Authentication*.
- 2) *Dashboard* yang menggunakan data sensor untuk menunjukkan status kapasitas tempat sampah.
- 3) Notifikasi otomatis yang memberikan peringatan ketika volume sampah melebihi batas tertentu.
- 4) Halaman riwayat aktivitas, di mana dapat melihat berapa kapasitas sebelum dikosongkan dan kondisi kapasitas setelah dibersihkan.
- 5) Halaman Tambah Tempat Sampah digunakan untuk menambahkan data tempat sampah baru.

- 6) Halaman pengaturan akun digunakan untuk menampilkan informasi pengguna dan memilih untuk keluar (*logout*).

d. Pengujian Internal

Pengujian internal menggunakan metode *Black Box Testing* setelah seluruh fitur utama aplikasi berhasil digunakan. Tujuan dari tahap ini adalah untuk memastikan bahwa semua operasi sistem berjalan sesuai dengan kebutuhan dan bahwa proses integrasi aplikasi dengan basis data dan perangkat sensor IoT tidak mengalami kesalahan.

Pengujian dilakukan dengan memberikan berbagai skenario *input* dan melihat hasilnya untuk mengevaluasi kesesuaian antara hasil yang sebenarnya dan yang diharapkan. Proses *login* dan registrasi pengguna, pengiriman notifikasi otomatis, pembaruan data kapasitas tempat sampah secara *real-time*, dan penyimpanan riwayat aktivitas di *database real-time* Firebase adalah beberapa fitur yang diuji.

Hasil pengujian internal ini menunjukkan bahwa sistem beroperasi dengan baik dan bahwa semua bagian aplikasi *mobile*, *database*, dan sensor IoT telah terhubung secara terintegrasi seperti yang direncanakan pada awalnya.

e. Evaluasi

Sebelum diuji oleh pengguna akhir, tahap evaluasi *prototype* dilakukan setelah sistem melewati proses pengujian internal untuk memastikan bahwa setiap fungsi utama aplikasi berjalan sesuai dengan kebutuhan. Evaluasi ini bertujuan untuk menilai kinerja sistem dalam hal fungsionalitas, integrasi sensor IoT, tampilan antarmuka, dan kemudahan penggunaan awal. Proses evaluasi dilakukan seperti berikut:

1. Meninjau hasil pengujian internal (*Blackbox Testing*), untuk memastikan bahwa tidak ada kesalahan logika atau tampilan dalam aplikasi.
2. Mengawasi stabilitas koneksi IoT dan sensor Firebase untuk memastikan data kapasitas tempat sampah selalu diperbarui.
3. Untuk mempersiapkan perbaikan sebelum uji pengguna, catat hasil pengujian seperti tampilan yang tidak responsif, ukuran teks, atau keterlambatan pembaruan data.

Hasil dari tahap evaluasi ini akan digunakan untuk menentukan apakah aplikasi sudah siap untuk diuji oleh petugas kebersihan pada tahap pengujian eksternal, atau *feedback* pengguna. Selain itu, evaluasi ini akan berfungsi sebagai dasar untuk menilai tingkat kesiapan sistem sebelum tahap selanjutnya, sebagai dasar kesiapan sistem sebelum dilakukan pengujian kepada pengguna pada tahap selanjutnya untuk mengukur tingkat kemudahan penggunaan aplikasi.

### 3.5 *Deployment Delivery & Feedback (Penerapan dan Umpan Balik)*

Tahap terakhir dari metode *prototype* adalah penerapan dan umpan balik. Pada tahap ini, aplikasi diuji secara langsung oleh pengguna akhir setelah melalui tahap pengujian internal dan evaluasi. Tujuan utama dari tahap ini adalah mengumpulkan tanggapan pengguna tentang kemudahan penggunaan (*usability*), tampilan, dan kesesuaian fitur aplikasi dengan kebutuhan operasional di lapangan (Pressman, 2010).

Sebelum aplikasi diuji oleh pengguna akhir, pengujian internal menggunakan *Black Box* dilakukan untuk memastikan bahwa semua fungsi penting bekerja sesuai dengan kebutuhan. Pengujian ini hanya menguji *input* dan *output* tanpa melihat struktur kode, sehingga cocok untuk menguji perilaku sistem dari perspektif pengguna.

#### 3.5.1 *Pengujian Black Box Testing*

Pengujian dilakukan pada beberapa fitur utama aplikasi seperti proses *login*, memantau kapasitas tempat sampah secara *real-time*, menerima notifikasi otomatis, dan mencatat riwayat aktivitas mereka. Setelah uji coba selesai menunjukkan bahwa sistem berjalan stabil dan layak diuji oleh pengguna akhir hasil pengujian menunjukkan bahwa fungsi utama sistem dapat berjalan sesuai dengan skenario pengujian dan siap untuk diuji oleh pengguna akhir. Berikut tabel pengujian:

Tabel 3.3 Pengujian *Black Box Testing*

| No | Halaman/<br>Fitur | Skenario pengujian                       | Input                               | hasil yang diharapkan                     |
|----|-------------------|------------------------------------------|-------------------------------------|-------------------------------------------|
| 1. | Registrasi        | Mengisi data registrasi dengan benar     | Data berhasil disimpan              | Masuk ke halaman Login                    |
| 2. | Login             | Email & Password benar                   | Email valid, password benar         | Masuk ke <i>dashboard</i>                 |
| 3. | Login             | Email/Password salah                     | Email/ password salah               | Muncul peringatan “email/password salah”  |
| 4. | Dashboard         | Menampilkan data kapasitas tempat sampah | Data kapasitas dari <i>database</i> | Status kapasitas sesuai data              |
| 5. | Dashboard         | Pembaruan kapasitas tempat sampah        | Data kapasitas berubah              | Indikator kapasitas diperbaharui otomatis |
| 6. | Notifikasi        | Kapasitas melebihi ambang batas          | Data kapasitas > batas              | Notifikasi otomatis muncul                |

|     |                      |                                     |                                                             |                                       |
|-----|----------------------|-------------------------------------|-------------------------------------------------------------|---------------------------------------|
| 7.  | Riwayat              | Penambahan aktivitas pembersihan    | Data aktivitas baru                                         | Riwayat bertambah                     |
| 8.  | Riwayat              | Menampilkan riwayat pembersihan     | Data aktivitas tersimpan                                    | Data riwayat tampil lengkap           |
| 9.  | Pengaturan akun      | Melakukan <i>logout</i>             | Tekan tombol <i>logout</i>                                  | Keluar dari aplikasi                  |
| 10. | Tambah Tempat Sampah | Menambahkan data tempat sampah baru | Nama tempat sampah, jenis perangkat, dan <i>path</i> sensor | Data tempat sampah berhasil tersimpan |

### 3.5.2 Pengujian *System Usability Scale* (SUS)

Setelah selesai pengujian, responden memberikan umpan balik melalui wawancara singkat dan kuesioner. Evaluasi *usability* menggunakan metode *System Usability Scale* (SUS). Kuesioner SUS terdiri dari sepuluh pernyataan yang dinilai berdasarkan skala Likert dari 1-5, yang menunjukkan persetujuan terhadap pernyataan dalam kuesioner. Setiap pernyataan dalam kuesioner SUS memiliki penilaian yang disesuaikan berdasarkan nomor pernyataan ganjil dan genap.

Tabel 3.4 Pengujian *System Usability Scale* (SUS)

| Skala/Bobot | Keterangan          |
|-------------|---------------------|
| 1           | Sangat Tidak Setuju |
| 2           | Tidak Setuju        |
| 3           | Netral              |
| 4           | Setuju              |
| 5           | Sangat Setuju       |

Berikut ini adalah beberapa pertanyaan yang mencakup elemen penting terkait dengan evaluasi *usability* menggunakan *System Usability Scale* (SUS), untuk memberikan gambaran lebih lanjut tentang kuesioner yang digunakan dalam penelitian ini:

Tabel 3.5 Pertanyaan *System Usability Scale* (SUS)

| No. | Pertanyaan                                                                                      | Skala |
|-----|-------------------------------------------------------------------------------------------------|-------|
| 1.  | Saya merasa bahwa saya akan sering menggunakan aplikasi ini                                     | 1-5   |
| 2.  | Saya merasa aplikasi ini terlalu rumit                                                          | 1-5   |
| 3.  | Saya merasa aplikasi ini mudah digunakan                                                        | 1-5   |
| 4.  | Saya merasa membutuhkan bantuan teknis untuk dapat menggunakan aplikasi ini                     | 1-5   |
| 5.  | Saya merasa fitur-fitur pada aplikasi ini berjalan sesuai fungsi                                | 1-5   |
| 6.  | Saya merasa terdapat terlalu banyak ketidaksesuaian dalam aplikasi ini                          | 1-5   |
| 7.  | Saya merasa sebagian besar orang akan belajar menggunakan aplikasi ini dengan cepat             | 1-5   |
| 8.  | Saya merasa aplikasi ini membingungkan untuk digunakan                                          | 1-5   |
| 9.  | Saya merasa percaya diri saat menggunakan aplikasi ini                                          | 1-5   |
| 10. | Saya merasa perlu mempelajari banyak hal terlebih dahulu sebelum dapat menggunakan aplikasi ini | 1-5   |

Ketika sudah menerima data yang dikumpulkan dari responden, langkah selanjutnya adalah menghitungnya. Untuk menghitung skor SUS, beberapa aturan harus diikuti. Berikut adalah aturan yang harus diperhatikan saat menghitung skor SUS:

- Untuk setiap pertanyaan bernomor ganjil, skor yang diberikan oleh pengguna dikurangi 1.
- Untuk setiap pertanyaan bernomor genap, skor diperoleh dengan 5 dikurangi skor pengguna.
- Skor SUS dihitung dengan menjumlahkan semua skor pertanyaan dan kemudian mengalikan hasilnya dengan 2,5.

Rumus perhitungan skor SUS dapat dibuat berdasarkan aturan ini:

$$\text{Skor SUS} = ((Q1-1)+(5-Q2)+(Q3-1)+(5-Q4)+(Q5-1)+(5-Q6)+(Q7-1)+(5-Q8)+(Q9-1)+(5-Q10)) \times 2,5$$

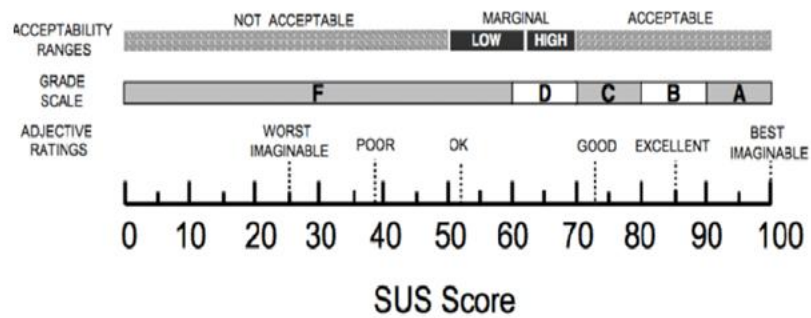
Gambar 3.14 Rumus Skor SUS

Tiga komponen umum, penerimaan, skala nilai, dan penilaian kata sifat, dapat digunakan untuk menentukan hasil perhitungan. Skala nilai SUS digunakan sebagai acuan untuk menginterpretasikan tingkat *usability* aplikasi pada tahap pengujian pengguna. Untuk skor SUS, setiap tingkat nilai memiliki bobot yang ditentukan.

- Tingkat "A" = skor lebih dari 80,3 dan dianggap sebagai "Excellent".
- Tingkat "B" = skor antara 68 dan 80,3 dan dianggap sebagai "Good".

- c. Tingkat "C" = skor 68 dan dianggap sebagai "OK".
- d. Tingkat "D" = skor antara 51 dan 67 dan dianggap sebagai "Poor".
- e. Tingkat "F" = skor di bawah 51, dan dianggap sebagai "Awful" (Ratnawati & Anandito, 2024).

Berikut adalah kriteria penilaian dan penjelasan lebih lanjut mengenai skala nilai yang digunakan:



Gambar 3.15 Skor SUS

## **BAB IV**

### **HASIL dan PEMBAHASAN**

#### **4.1 Implementasi Sistem**

Aplikasi Tempat Sampah Cerdas Berbasis *Mobile* telah diimplementasikan sesuai dengan rancangan yang dijelaskan dalam Bab sebelumnya. Implementasi sistem dilakukan menggunakan *framework* Flutter, autentikasi pengguna Firebase, dan Database Realtime Firebase untuk penyimpanan data kapasitas tempat sampah.

##### **4.1.1 Halaman Login dan Registrasi**

Bagian pertama dari aplikasi Tempat Sampah Cerdas Berbasis *Mobile* adalah halaman *login* dan registrasi, di mana pengguna diminta untuk mengisi informasi pribadi mereka, yaitu alamat email dan kata sandi, yang kemudian disimpan ke dalam sistem melalui autentikasi Firebase, yang digunakan untuk memastikan bahwa hanya pengguna yang terdaftar di aplikasi yang dapat mengaksesnya.

Halaman *login* berfungsi untuk memastikan bahwa informasi pengguna yang telah terdaftar benar. Sistem akan memeriksa email dan kata sandi yang dimasukkan ke Firebase untuk memastikan bahwa data sesuai. Jika data sesuai, pengguna akan diarahkan ke halaman *dashboard*, jika tidak, akan muncul pesan kesalahan. Implementasi halaman ini memastikan bahwa akses ke aplikasi tetap aman dan memastikan bahwa pihak yang berwenang dapat menggunakan sistem.

The image displays two mobile application screens side-by-side. The left screen is the login page, featuring a green header, a 'Selamat Datang!' (Welcome!) greeting with a globe icon, and input fields for 'Email' and 'Kata Sandi' (password) with a 'Masuk' (Login) button. The right screen is the registration page, titled 'Buat Akun' (Create Account) with the subtitle 'Masukkan Data Pribadi Anda' (Enter your personal data). It includes input fields for 'Nama Depan' (First Name), 'Nama Belakang' (Last Name), 'Email', 'Nomor Telepon' (Phone Number), 'Kata Sandi' (password), and 'Konfirmasi Kata Sandi' (confirm password), followed by a 'Buat Akun' button and a link to 'Masuk' (Login) for existing users.

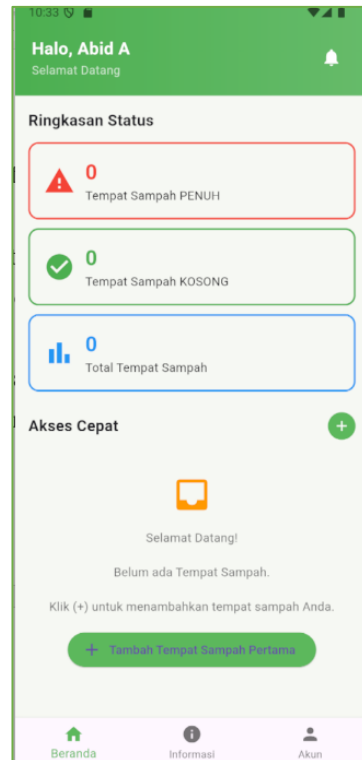
Gambar 4.1 Halaman *Login dan Registrasi*

#### 4.1.2 Halaman Dashboard

Halaman *dashboard* aplikasi adalah halaman utama yang menampilkan informasi kondisi tempat sampah. Ini menampilkan status kapasitas tempat sampah dalam bentuk indikator kondisi seperti kosong, setengah penuh, dan penuh. Informasi ini diperbarui secara *real-time* dari *database* data waktu nyata Firebase.

Pada saat pengguna baru melakukan pendaftaran, halaman *dashboard* belum menampilkan data tempat sampah karena belum ada tempat sampah yang terdaftar pada sistem. Oleh karena itu, pengguna perlu menambahkan data tempat sampah terlebih dahulu melalui fitur tambah tempat sampah. Setelah data tempat sampah berhasil ditambahkan dan terhubung dengan sensor, informasi kondisi tempat sampah akan muncul pada *dashboard* sesuai dengan data yang diterima oleh sistem.

*Dashboard* dibuat untuk memberikan gambaran kondisi tempat sampah yang cepat dan mudah dipahami oleh pengguna. Halaman ini memungkinkan petugas kebersihan untuk menentukan prioritas penanganan tempat sampah berdasarkan kondisi yang ditampilkan oleh sistem.

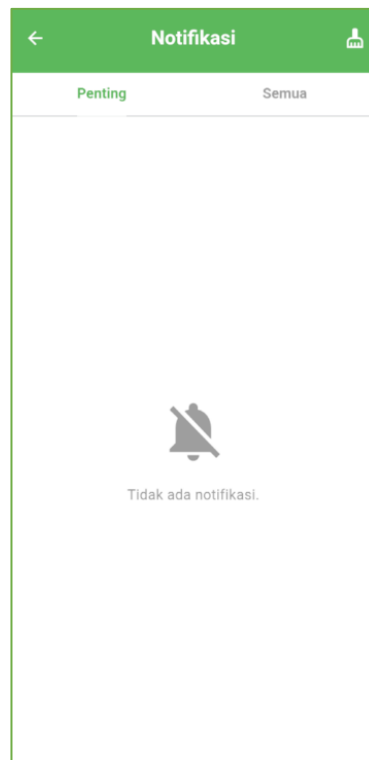


Gambar 4.2 Halaman *Dashboard*

#### 4.1.3 Halaman Notifikasi

Halaman notifikasi berfungsi untuk menampilkan pesan peringatan kepada pengguna ketika kapasitas tempat sampah telah mencapai batas tertentu. Pesan peringatan ini dibuat berdasarkan perubahan data kapasitas yang diterima oleh sistem dan diproses secara otomatis oleh sistem.

Tujuan dari halaman notifikasi adalah untuk membantu pengguna merespon kondisi tempat sampah dengan lebih cepat tanpa harus mengeceknya secara manual melalui *dashboard*. Keterangan kondisi tempat sampah yang memerlukan tindakan segera ditampilkan pada halaman ini.



Gambar 4.3 Halaman Notifikasi

#### 4.1.4 Halaman Riwayat

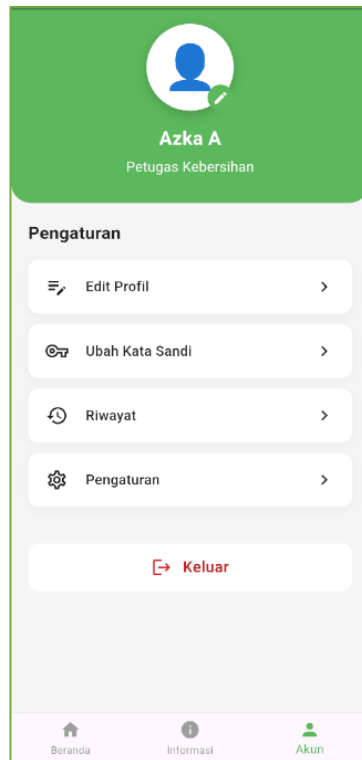
Halaman riwayat digunakan untuk menampilkan catatan aktivitas pembersihan tempat sampah, serta kondisi kapasitas terakhir sebelum pembersihan dilakukan. Data riwayat disimpan dalam Firebase Realtime Database dan ditampilkan kembali secara terstruktur dalam aplikasi. Dengan halaman riwayat, pengguna dapat melihat apa yang telah dilakukan dan menggunakannya sebagai contoh pengelolaan tempat sampah.



Gambar 4.4 Halaman Riwayat

#### 4.1.5 Halaman Pengaturan

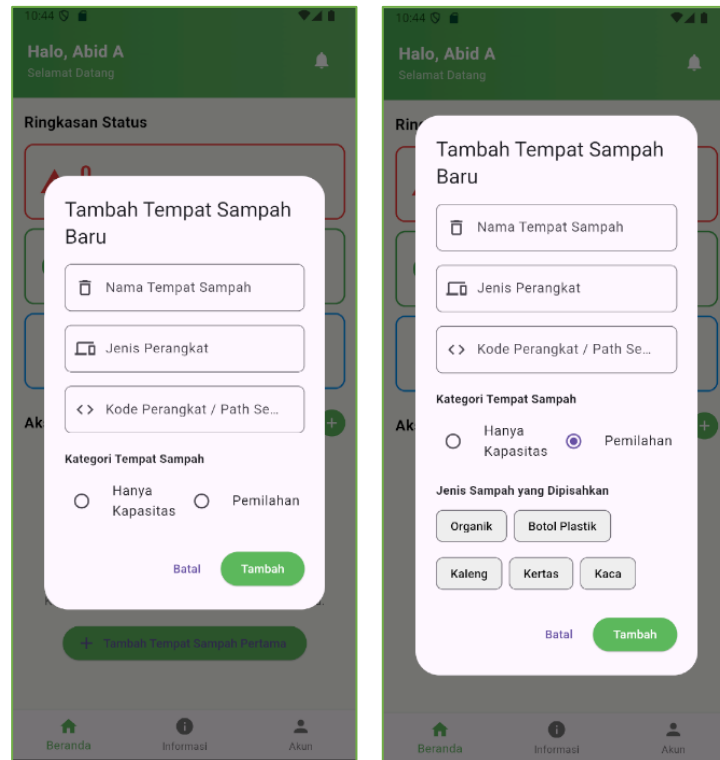
Halaman akun berfungsi sebagai pusat pengelolaan informasi dan aktivitas pengguna pada aplikasi Tempat Sampah Cerdas Berbasis *Mobile*. Pada halaman ini ditampilkan informasi profil pengguna, seperti nama dan peran serta beberapa menu, antara lain edit profil untuk memperbarui data pengguna, ubah kata sandi untuk menjaga keamanan akun, riwayat untuk melihat aktivitas pembersihan, serta menu pengaturan untuk pengaturan tambahan aplikasi. Selain itu, tersedia tombol keluar (*logout*) yang digunakan untuk mengakhiri sesi penggunaan aplikasi dan mengarahkan pengguna kembali ke halaman *login*.



Gambar 4.5 Halaman Pengaturan

#### 4.1.6 Halaman Tambah Tempat Sampah

Pada sistem yang dikembangkan, penambahan data tempat sampah dilakukan melalui fitur tambah tempat sampah pada aplikasi *mobile*. Pengguna dapat mengisi data berupa nama tempat sampah, jenis perangkat, serta kode perangkat atau *path sensor*. Setelah data disimpan, sistem akan menambahkan data tersebut ke Firebase Realtime. Fitur tambah tempat sampah hanya digunakan untuk mendaftarkan data tempat sampah ke aplikasi. Data kapasitas tidak otomatis muncul jika belum ada sensor atau perangkat IoT yang mengirimkan data ke Firebase.



Gambar 4.6 Halaman Tambah Tempat Sampah

## 4.2 Integrasi dengan *Internet of Things* (IoT)

Sistem Tempat Sampah Cerdas Berbasis *Mobile* pada penelitian ini terintegrasi dengan dua jenis perangkat tempat sampah berbasis *Internet of Things* (IoT) yang memiliki fungsi berbeda. Perbedaan fungsi ini dirancang untuk menyesuaikan kebutuhan pengelolaan sampah di lingkungan Fakultas Teknologi Industri.

Tempat sampah pertama berfungsi sebagai tempat sampah pemantauan kapasitas, yang hanya mendeteksi tingkat kepenuhan tempat sampah tanpa melakukan proses pemilahan sampah. Sensor pada tempat sampah ini membaca volume sampah yang masuk dan mengirimkan data kapasitas ke Firebase Realtime Database. Data tersebut kemudian ditampilkan pada aplikasi *mobile* dalam bentuk persentase dan status kapasitas.

Tempat sampah kedua merupakan tempat sampah pemilah sampah, yang memiliki kemampuan untuk memisahkan sampah ke dalam beberapa kategori, yaitu sampah organik, kaleng, dan botol plastik. Selain melakukan pemilahan, tempat sampah ini juga dilengkapi dengan sensor untuk memantau kapasitas masing-masing jenis sampah. Data hasil pemilahan dan kapasitas setiap jenis sampah dikirimkan ke Firebase Realtime Database secara terpisah.

Firebase Realtime Database berperan sebagai media penghubung antara perangkat IoT dan aplikasi *mobile*. Aplikasi secara berkala membaca data dari kedua tempat sampah tersebut dan

menampilkannya pada halaman *dashboard*. Dengan adanya pemisahan data berdasarkan jenis tempat sampah dan kategori sampah, pengguna dapat memantau kondisi tempat sampah secara lebih detail dan terstruktur.

Apabila kapasitas salah satu tempat sampah atau salah satu kategori sampah pada tempat sampah pemilah telah melewati ambang batas yang ditentukan, sistem akan memicu pengiriman notifikasi kepada pengguna. Integrasi ini memungkinkan petugas kebersihan untuk mengetahui jenis tempat sampah mana yang perlu segera ditangani, sehingga proses pengelolaan sampah dapat dilakukan secara lebih efisien dan tepat sasaran.

### 4.3 Hasil Pengujian

Hasil pengujian ini menunjukkan bagaimana aplikasi diuji pada sejumlah responden untuk mengetahui apakah sistem dapat berjalan sesuai dengan fungsi yang telah dirancang pada bab sebelumnya.

#### 4.3.1 Hasil Pengujian *Black Box Testing*

Pengujian *Black Box Testing* dilakukan untuk menguji *input* dan *output* aplikasi tanpa perlu pengetahuan tentang bagaimana aplikasi dibuat. Pengujian ini menggunakan berbagai skenario pengguna dievaluasi untuk memastikan aplikasi berfungsi sesuai dengan yang diharapkan. Hasil dari pengujian ini dapat dilihat pada tabel berikut.

Tabel 4.1 Hasil Pengujian Black Box Testing

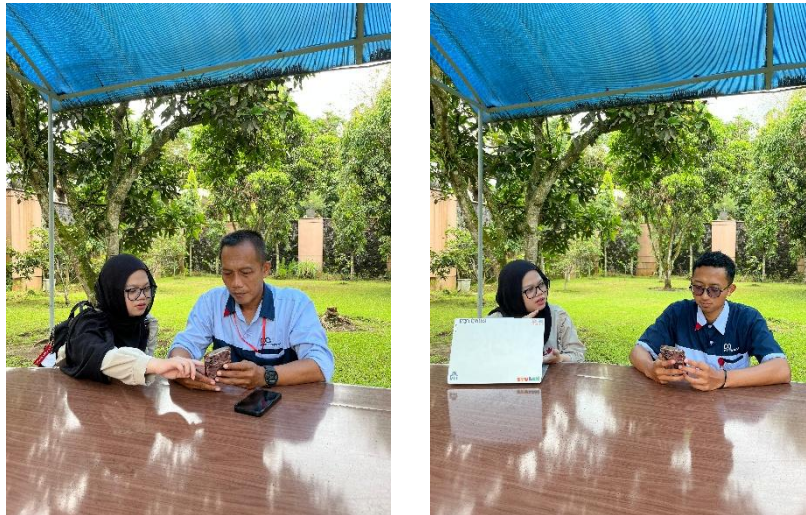
| No | Halaman/Fitur    | Skenario pengujian                       | Input                                      | hasil yang diharapkan                            | Hasil pengujian |
|----|------------------|------------------------------------------|--------------------------------------------|--------------------------------------------------|-----------------|
| 1  | Registrasi       | Mengisi data registrasi dengan benar     | Data berhasil disimpan                     | Masuk ke halaman <i>Login</i>                    | Berhasil        |
| 2  | <i>Login</i>     | Email & <i>Password</i> benar            | Email <i>valid</i> , <i>password</i> benar | Masuk ke <i>dashboard</i>                        | Berhasil        |
| 3  | <i>Login</i>     | Email/ <i>password</i> salah             | Email/ <i>password</i> salah               | Muncul peringatan “email/ <i>password</i> salah” | Berhasil        |
| 4  | <i>Dashboard</i> | Menampilkan data kapasitas tempat sampah | Data kapasitas dari <i>database</i>        | Status kapasitas sesuai data                     | Berhasil        |

|    |                      |                                   |                                        |                                           |          |
|----|----------------------|-----------------------------------|----------------------------------------|-------------------------------------------|----------|
| 5  | Dashboard            | Pembaruan kapasitas tempat sampah | Data kapasitas berubah                 | Indikator kapasitas diperbaharui otomatis | Berhasil |
| 6  | Notifikasi           | Kapasitas melebihi ambang batas   | Data kapasitas > batas                 | Notifikasi otomatis muncul                | Berhasil |
| 7  | Riwayat              | Penambahan aktivitas pembersihan  | Data aktivitas baru                    | Riwayat bertambah                         | Berhasil |
| 8  | Riwayat              | Menampilkan riwayat pembersihan   | Data aktivitas tersimpan               | Data riwayat tampil lengkap               | Berhasil |
| 9  | Pengaturan akun      | Melakukan <i>logout</i>           | Tekan tombol <i>logout</i>             | Keluar dari aplikasi                      | Berhasil |
| 10 | Tambah Tempat Sampah | Menambahkan tempat sampah baru    | Jenis perangkat dan <i>path sensor</i> | Data tersimpan & tampil di dashboard      | Berhasil |

Hasil pengujian menunjukkan bahwa seluruh fungsi aplikasi berjalan sesuai rancangan dan dapat digunakan dengan baik oleh pengguna.

#### 4.3.2 Hasil Pengujian *System Usability Scale* (SUS)

Pengujian *usability* dilakukan menggunakan metode *System Usability Scale* (SUS). Sebanyak 10 responden diminta menggunakan aplikasi dan mengisi kuesioner yang berisi 10 pernyataan dengan skala Likert 1–5. Dokumentasi kegiatan pengujian *usability* dengan calon pengguna aplikasi dapat dilihat pada gambar berikut.



Gambar 4.7 Pengujian Bersama Pengguna

Setelah pengguna mencoba aplikasi, mereka diminta mengisi kuesioner yang terdiri dari 10 pertanyaan SUS yang ditunjukkan pada. Hasil dari SUS dapat dilihat pada table.

Tabel 4.2 Hasil Pengujian *System Usability Scale* (SUS)

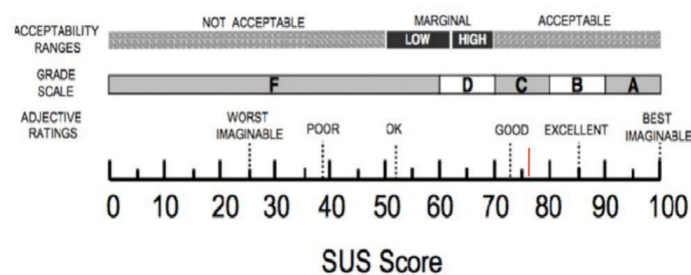
| Responden | Skor |    |    |    |    |    |    |    |    |     |
|-----------|------|----|----|----|----|----|----|----|----|-----|
|           | P1   | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
| R1        | 5    | 1  | 5  | 1  | 4  | 1  | 5  | 1  | 5  | 1   |
| R2        | 5    | 3  | 5  | 3  | 5  | 1  | 5  | 3  | 5  | 1   |
| R3        | 5    | 5  | 5  | 5  | 5  | 2  | 5  | 1  | 5  | 3   |
| R4        | 5    | 1  | 5  | 1  | 5  | 1  | 5  | 1  | 5  | 1   |
| R5        | 5    | 5  | 4  | 5  | 4  | 4  | 5  | 4  | 5  | 5   |
| R6        | 5    | 1  | 5  | 4  | 4  | 3  | 5  | 1  | 5  | 4   |
| R7        | 1    | 5  | 5  | 3  | 5  | 3  | 5  | 1  | 5  | 5   |
| R8        | 4    | 2  | 4  | 2  | 4  | 2  | 4  | 2  | 4  | 2   |
| R9        | 5    | 1  | 5  | 3  | 5  | 1  | 5  | 1  | 5  | 1   |
| R10       | 3    | 3  | 3  | 2  | 2  | 3  | 2  | 2  | 3  | 2   |

Berdasarkan data yang didapat di atas, dilakukan perhitungan menggunakan rumus yang sudah dijelaskan pada bab III untuk memperoleh hasil akhir. Hasil perhitungan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Akhir Pengujian *System Usability Scale* (SUS)

| Responden | Skor |    |    |    |    |    |    |    |    |     | Jumlah | Jumlah*2,5 |
|-----------|------|----|----|----|----|----|----|----|----|-----|--------|------------|
|           | P1   | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |        |            |
| R1        | 4    | 4  | 4  | 4  | 3  | 4  | 4  | 4  | 4  | 4   | 39     | 97.5       |
| R2        | 4    | 2  | 4  | 2  | 4  | 4  | 4  | 2  | 4  | 4   | 34     | 85         |
| R3        | 4    | 0  | 4  | 0  | 4  | 3  | 4  | 4  | 4  | 2   | 29     | 72.5       |
| R4        | 4    | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 4   | 40     | 100        |
| R5        | 4    | 0  | 3  | 0  | 3  | 1  | 4  | 1  | 4  | 0   | 20     | 50         |
| R6        | 4    | 4  | 4  | 1  | 3  | 2  | 4  | 4  | 4  | 1   | 31     | 77.5       |
| R7        | 0    | 0  | 4  | 2  | 4  | 2  | 4  | 4  | 4  | 0   | 24     | 60         |
| R8        | 3    | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3   | 30     | 75         |
| R9        | 4    | 4  | 4  | 2  | 4  | 4  | 4  | 4  | 4  | 4   | 38     | 95         |
| R10       | 2    | 2  | 2  | 3  | 1  | 2  | 1  | 3  | 2  | 3   | 21     | 52.5       |
| Rata-rata |      |    |    |    |    |    |    |    |    |     |        | 76.5       |

Berdasarkan hasil perhitungan, diperoleh nilai rata-rata *System Usability Scale* sebesar 76,5. visualisasi hasil pengujian dapat dilihat pada Gambar 4.8.

Gambar 4.8 Hasil Skor *System Usability Scale* (SUS)

#### 4.3.3 Hasil Observasi Pengujian

Pengujian alat dan aplikasi dilakukan untuk mengetahui sejauh mana sistem tempat sampah cerdas dapat berfungsi secara optimal ketika digunakan secara langsung oleh pengguna. Pengujian ini melibatkan 30 mahasiswa dan petugas kebersihan yang diminta untuk mencoba alat secara langsung di lingkungan kampus. Setiap responden diminta untuk memasukkan sampah ke dalam tempat sampah cerdas, kemudian mengamati proses deteksi yang dilakukan oleh sensor, serta melihat hasil yang ditampilkan pada aplikasi *mobile*.

Selama proses pengujian, responden mengikuti alur penggunaan mulai dari interaksi dengan alat hingga melihat hasil monitoring pada aplikasi. Pengujian dilakukan dengan skenario penggunaan langsung, di mana setiap responden mencoba memasukkan beberapa jenis sampah ke dalam alat untuk melihat kemampuan sistem dalam mendeteksi dan mengklasifikasikan sampah. Responden juga diminta untuk mengamati respon aplikasi *mobile* terhadap data yang dikirimkan oleh perangkat, termasuk kecepatan pembaruan data dan keakuratan informasi yang ditampilkan.

Berdasarkan hasil pengujian yang telah dilakukan, dari 30 responden terdapat 25 responden yang berhasil menggunakan alat dengan baik, di mana proses deteksi sampah dan penampilan data pada aplikasi berjalan sesuai dengan yang diharapkan. Sementara itu, lima responden lainnya mengalami kegagalan dalam proses penggunaan alat. Dengan demikian, diperoleh tingkat keberhasilan sebesar 83,3% dan tingkat kegagalan sebesar 16,7%.

Kegagalan yang terjadi selama pengujian umumnya disebabkan oleh beberapa faktor teknis yang berkaitan dengan keterbatasan sensor dan kamera yang digunakan. Dalam beberapa kondisi, sensor atau kamera tidak mampu mendeteksi objek sampah secara optimal, terutama ketika posisi sampah tidak berada pada area deteksi yang tepat. Selain itu, terdapat beberapa jenis sampah yang tidak termasuk dalam data atau kategori yang telah diprogram pada sistem, sehingga sensor dan kamera tidak dapat mengidentifikasi sampah tersebut ke dalam jenis tertentu. Kondisi ini menyebabkan sistem tidak memberikan hasil klasifikasi yang sesuai. Faktor lain yang mempengaruhi kegagalan adalah ukuran dan karakteristik sampah. Sampah dengan ukuran yang terlalu kecil atau memiliki bentuk yang tipis cenderung sulit dideteksi oleh sensor maupun kamera, sehingga proses identifikasi menjadi tidak akurat. Di samping itu, pada beberapa percobaan juga ditemukan adanya keterlambatan dalam pengiriman data dari perangkat ke aplikasi, sehingga hasil tidak langsung muncul secara *real-time*.

Dari sisi pengguna, sebagian besar responden (sekitar 80%) memberikan tanggapan yang positif terhadap sistem yang dikembangkan. Responden menyatakan bahwa aplikasi mudah digunakan dan informasi yang ditampilkan cukup jelas untuk membantu memahami kondisi tempat sampah. Selain itu, fitur pemilahan sampah dinilai memberikan manfaat tambahan karena membantu proses pengelolaan sampah menjadi lebih terarah. Namun demikian, responden juga memberikan masukan bahwa akurasi alat masih perlu ditingkatkan agar hasil deteksi dapat lebih konsisten dan minim kesalahan.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem telah mampu berjalan dengan cukup baik dan sesuai dengan tujuan penelitian, yaitu membantu proses monitoring dan pengelolaan sampah secara lebih efisien. Meskipun demikian, masih terdapat beberapa keterbatasan pada aspek perangkat keras, khususnya pada akurasi sensor dan kondisi lingkungan, sehingga diperlukan pengembangan lebih lanjut untuk meningkatkan performa sistem secara keseluruhan.

#### 4.4 Pembahasan

Berdasarkan hasil pengujian yang telah dilakukan, sistem tempat sampah cerdas menunjukkan tingkat keberhasilan sebesar 83,3% dalam mendeteksi dan menampilkan informasi sampah melalui aplikasi *mobile*. Hasil ini menunjukkan bahwa sistem yang dikembangkan telah mampu menjalankan fungsi utamanya, yaitu melakukan monitoring kapasitas dan mendukung proses pengelolaan sampah secara lebih efisien dibandingkan metode manual.

Tingkat keberhasilan tersebut dipengaruhi oleh integrasi antara perangkat IoT dan aplikasi *mobile* yang berjalan dengan cukup baik. Data yang dikirimkan oleh sensor dapat ditampilkan pada aplikasi secara relatif cepat dan akurat, sehingga pengguna dapat mengetahui kondisi tempat sampah secara *real-time*. Hal ini sejalan dengan penelitian sebelumnya yang menyatakan bahwa penerapan teknologi *Internet of Things* (IoT) mampu meningkatkan efisiensi dalam proses monitoring lingkungan karena memungkinkan pengiriman data secara otomatis dan berkelanjutan.

Namun demikian, masih terdapat tingkat kegagalan sebesar 16,7% yang menunjukkan adanya keterbatasan pada sistem. Kegagalan ini umumnya disebabkan oleh faktor teknis, seperti keterbatasan sensor dan kamera dalam mendeteksi objek sampah, terutama pada kondisi tertentu. Sampah dengan ukuran kecil atau tipis cenderung sulit terdeteksi, serta adanya jenis sampah yang tidak termasuk dalam data yang telah diprogram menyebabkan sistem tidak dapat mengklasifikasikan objek dengan tepat. Selain itu, faktor lingkungan seperti pencahayaan juga mempengaruhi akurasi deteksi, serta adanya keterlambatan dalam pengiriman data yang berdampak pada tampilan informasi yang tidak sepenuhnya *real-time*.

Dari sisi pengguna, hasil evaluasi menunjukkan bahwa mayoritas responden memberikan tanggapan positif terhadap sistem yang dikembangkan. Pengguna menilai bahwa aplikasi mudah digunakan dan informasi yang ditampilkan cukup jelas untuk membantu proses monitoring. Hal ini menunjukkan bahwa aspek *usability* dari aplikasi telah terpenuhi, di mana pengguna dapat memahami dan menggunakan sistem tanpa kesulitan yang berarti. Temuan ini juga didukung oleh hasil pengujian *System Usability Scale* (SUS) yang menunjukkan bahwa aplikasi berada pada kategori yang dapat diterima oleh pengguna.

Meskipun sistem telah menunjukkan performa yang cukup baik, hasil penelitian ini juga menegaskan bahwa masih diperlukan pengembangan lebih lanjut, khususnya pada aspek perangkat keras dan akurasi deteksi. Peningkatan kualitas sensor, perluasan dataset jenis

sampah, serta optimasi proses pengiriman data menjadi faktor penting untuk meningkatkan tingkat keberhasilan sistem di masa mendatang.

Secara keseluruhan, sistem tempat sampah cerdas berbasis *mobile* yang dikembangkan telah mampu memberikan solusi terhadap permasalahan monitoring sampah di lingkungan kampus. Dibandingkan dengan metode manual, sistem ini memberikan kemudahan dalam pemantauan kondisi tempat sampah secara *real-time*, sehingga dapat meningkatkan efisiensi kerja petugas kebersihan dan mendukung pengelolaan lingkungan yang lebih baik.

## **BAB V**

### **KESIMPULAN dan SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil dari implementasi pengembangan dan pengujian aplikasi tempat sampah cerdas berbasis *mobile* yang telah dilakukan pada bab sebelumnya, dapat ditarik kesimpulan sebagai berikut:

- a. Pengembangan aplikasi Tempat Sampah Cerdas Berbasis *Mobile* ini menggunakan metode *Prototyping*. Proses pengembangan sistem dilakukan secara bertahap melalui lima tahapan utama, yaitu komunikasi (*communication*), perencanaan cepat (*quick plan*), desain cepat (*modeling quick design*), pembangunan prototipe (*construction of prototype*), dan penerapan serta umpan balik (*deployment delivery & feedback*). Metode ini memungkinkan keterlibatan pengguna secara langsung dalam setiap tahapan, yang meningkatkan akurasi pengembangan sistem sesuai dengan kebutuhan pengguna.
- b. Berdasarkan hasil *black-box testing*, aplikasi berfungsi dengan baik dan sesuai dengan skenario yang ditentukan. Semua fitur, seperti *login*, pemantauan kapasitas tempat sampah, pengiriman notifikasi otomatis, penambahan tempat sampah, dan pengelolaan riwayat aktivitas, dapat berfungsi dengan baik. Pengujian *System Usability Scale* (SUS) menunjukkan skor rata-rata 76.5, yang termasuk dalam kategori "Good". Sebagian besar responden memberikan tanggapan positif terhadap aplikasi, menyatakan bahwa aplikasi mudah digunakan dan fitur pemilahan sampah membantu pengelolaan sampah lebih efisien. Meskipun demikian, beberapa responden memberikan masukan untuk meningkatkan akurasi deteksi dan kecepatan pengiriman data.

#### **5.2 Saran**

Pengembangan aplikasi tempat sampah cerdas berbasis *mobile* yang telah dilakukan masih memiliki kekurangan yang masih dapat dikembangkan dan diperbaiki. Adapun saran yang dapat diberikan peneliti pada penelitian berikutnya yaitu:

- a. Pengiriman data antara perangkat dan aplikasi perlu dioptimalkan untuk mengurangi keterlambatan pembaruan informasi. Kecepatan pengiriman data harus ditingkatkan agar informasi kapasitas tempat sampah dapat diperbarui secara *real-time*.
- b. Aplikasi sebaiknya dikembangkan untuk mendeteksi lebih banyak jenis sampah, seperti sampah elektronik atau jenis sampah lainnya yang mungkin terlewatkan pada pengujian awal. Hal ini akan meningkatkan fungsionalitas sistem pemilahan sampah secara otomatis.

- c. Mengembangkan aplikasi untuk *platform* lain seperti iOS atau aplikasi berbasis web untuk memperluas jangkauan pengguna dan meningkatkan aksesibilitas aplikasi.

## DAFTAR PUSTAKA

- Agustino, R., Saputro, M. I., Gustiawan, H., Sakaria, M. A., & Widyahastuti, F. (2024). Inovasi pengelolaan sampah: Tempat sampah pintar berbasis IoT di Museum MH. Thamrin Jakarta. *Jurnal Teknologi Informatika dan Komputer*, 10(1), 309–325. <https://doi.org/10.37012/jtik.v10i1.2129>
- 'Aisy, R., Mursityo, Y. T., & Wijoyo, S. H. (2024). Evaluasi usability aplikasi mobile Sampangan menggunakan metode usability testing dan System Usability Scale (SUS). *Jurnal Teknologi Informasi dan Ilmu Komputer*, 11(1), 19–26. <https://doi.org/10.25126/jtiik.20241116613>
- Alda, M., Koto, M. H., & Wardani, A. (2023). Implementasi metode prototyping pada rancangan toko tanaman berbasis Android. *Rabit: Jurnal Teknologi dan Sistem Informasi Univrab*, 8(2), 254–261. <https://doi.org/10.36341/rabit.v8i2.3156>
- Alphita, A. P., & Saian, P. O. N. (2023). Pengembangan aplikasi edukasi pengelolaan sampah untuk anak sekolah berbasis mobile dengan teknologi machine learning. *IT-Explore: Jurnal Penerapan Teknologi Informasi dan Komunikasi*, 2(1), 1–17. <https://doi.org/10.24246/itexplore.v2i1.2023.pp1-17>
- Ammann, P., & Offutt, J. (2016). *Introduction to: Software testing* (2nd edn). Cambridge University Press. <https://doi.org/10.1017/9781316771273>
- Awaludin, A. W., Ashari, Aziz, A., Lubis, G. A. F., & Suryaningrat. (2023). Pengembangan mobile hybrid apps dengan Flutter. *Jurnal Cahaya Mandalika*, 3(2), 1433–1436.
- Bakhtiar, M. A., Naf'an, M. H., Rahmanda, A. A. D., Balqis, M., Ratnaningtyas, E. E., Setiawan, D., & Yunierra, N. E. (2023). Evaluasi pengelolaan sampah organik di kantin Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang. *Jurnal Analis*, 2(2), 209–217.

- Dzakiyyudin, M., & Frendiana, V. (2024). Perancangan aplikasi mobile “Bersih Plus” menggunakan framework Flutter untuk nasabah bank sampah. *Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika*, 4, 593–598. <https://doi.org/10.31284/p.snestik.2024.5914>
- Ernawati, S., & Rahayu, S. (2023). Analisa usability pada aplikasi identitas kependudukan digital menggunakan metode usability testing. *BIOS : Jurnal Teknologi Informasi dan Rekayasa Komputer*, 5(1), 12–19. <https://doi.org/10.37148/bios.v5i1.87>
- Fiqih, M., Nurul, & Syaiful, S. (2023). Penempatan bak sampah organik, anorganik, dan B3 dengan konsep go green Perumahan Budi Agung RW 03/RT 05. *Jurnal Pengabdian Masyarakat UIKA Jaya: SINKRON*, 1(2). <https://doi.org/10.32832/jpmuj.v1i2>
- Fitriana, G. F., Hashina, A., & Tanjung, N. A. F. (2022). Pengembangan aplikasi pengelolaan sampah berbasis Android studi kasus Bank Sampah Desa Kalibagor. *Journal of Dinda : Data Science, Information Technology, and Data Analytics*, 2(2), 97–102. <https://doi.org/10.20895/dinda.v2i2.741>
- Fridayanthie, E. W., Haryanto, H., & Tsabitah, T. (2021). Penerapan metode prototype pada perancangan sistem informasi penggajian karyawan (Persis Gawan) berbasis web. *Paradigma - Jurnal Komputer dan Informatika*, 23(2). <https://doi.org/10.31294/p.v23i2.10998>
- Ismail, M., Abdullah, R. K., & Abdussamad, S. (2021). Tempat sampah pintar berbasis Internet of Things (IoT) dengan sistem teknologi informasi. *Jambura Journal of Electrical and Electronics Engineering*, 3(1), 7–12. <https://doi.org/10.37905/jjee.v3i1.8099>
- Khairun Nisa Meiah Ngafidin, Arista, A., & Nisa Sofia Amriza, R. (2021). Implementasi Firebase Realtime Database pada aplikasi FeedbackMe sebagai penghubung guru dan orang tua. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 5(2), 327–334. <https://doi.org/10.29207/resti.v5i2.2909>

- Kholili, A. N. (2023). Sistem informasi pengelolaan sampah rumah tangga berbasis mobile. *INTECH*, 4(1), 28–34. <https://doi.org/10.54895/intech.v4i1.1982>
- Mamoun, R., Nasor, M., & Abulikailik, S. H. (2021). Design and development of mobile healthcare application prototype using Flutter. *2020 International Conference on Computer, Control, Electrical, and Electronics Engineering*. <https://doi.org/10.1109/ICCCEEE49695.2021.9429595>
- Marlina, A., Sari, A. N., Syahira, N. A., Syafarina, P., & Bintang, R. S. (2023). Edukasi mengenai pentingnya pemilahan serta pengolahan sampah untuk mengurangi dampak negatif terhadap lingkungan. *Darmabakti: Jurnal Inovasi Pengabdian dalam Penerbangan*, 4(1), 11–17. <https://doi.org/10.52989/darmabakti.v4i1.108>
- Maulana, M., Abyan Shidqi, M., Alifsyah Ramadhan, R. N., Arista Putri, S., Fadhlurokhman, R., Haykal Andana, M., Cholis Arethus, M., Khoirrosyid, H., Guritno, D., Mujiastuti, R., Rosanti, N., & Nurbaya Ambo, S. (2024). Tech talk: Software development training and framework utilization for website development. *Jurnal Pengabdian Kolaborasi dan Inovasi IPTEKS*, 2(1), 244–252. <https://doi.org/10.59407/jpki2.v2i1.508>
- Muhammad Helmi Satria Fedianto, Firza Prima Aditiawan, & Muhammad Muharrom Al Haromainy. (2023). Pengujian sistem jaringan dokumentasi dan informasi menggunakan black box testing dan white box testing. *Jurnal Publikasi Sistem Informasi dan Manajemen Bisnis*, 3(1), 213–221. <https://doi.org/10.55606/jupsim.v3i1.2447>
- Olethea Farah Utari, Humaini, I., & Ira Windarti. (2024). Pembuatan aplikasi penjemputan sampah anorganik (DTrash) menggunakan metode User Centered Design (UCD). *Jurnal Teknik dan Science*, 3(2), 137–146. <https://doi.org/10.56127/jts.v3i2.1559>

- Paksi, A. B., Hafidhoh, N., & Bimonugroho, S. K. (2023). Perbandingan model pengembangan perangkat lunak untuk proyek tugas akhir program vokasi. *Jurnal Masyarakat Informatika*, 14(1), 70–79. <https://doi.org/10.14710/jmasif.14.1.52752>
- Panjaitan, J., & Pakpahan, A. F. (2021). Perancangan sistem E-reporting menggunakan ReactJS dan Firebase. *Jurnal Teknik Informatika dan Sistem Informasi*, 7(1), 20–34. <https://doi.org/10.28932/jutisi.v7i1.3098>
- Pressman, R. S. (2010). *Software engineering: A practitioner's approach* (7th ed). McGraw-Hill.
- Putra, Y. W. S., Dawis, A. M., Novi, N., Natsir, F., Fitria, F., Widhiyanti, A. A. S., Hasan, F. N., Somantri, S., & Maniah, M. (2023). *Pengantar aplikasi mobile*. Penerbit Widina.
- Ramdhani, O., Yustiana, I., & Fergina, A. (2022). Rancang bangun sistem informasi bank sampah menggunakan metode prototype (studi kasus di Kampung Lembur Sawah, Sukabumi). *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 7(3), 757–767. <https://doi.org/10.29100/jipi.v7i3.3080>
- Ratnawati, S., & Anandito, W. D. (2024). Analisis usability pada aplikasi mobile banking BCA menggunakan System Usability Scale (SUS). *Jurnal Perangkat Lunak*, 6(2), 237–244. <https://doi.org/10.32520/jupel.v6i2.3285>
- Sekti, B. A., Ulfa, H., & Anwar, N. (2024). Prototype tempat sampah cerdas dengan monitoring berbasis IoT (studi kasus Universitas Esa Unggul Tangerang). *Jurnal Teknologi Informatika dan Komputer*, 10(1), 87–102. <https://doi.org/10.37012/jtik.v10i1.2014>
- Setiawan, P. R., Syefriani, & Wandri, R. (2025). Inovasi teknologi melalui pembelajaran Flutter: Menyongsong era aplikasi mobile. *Jurnal Sains dan Teknologi dalam Pengabdian Masyarakat*, 2(1), 1–9.
- Shabrina Ziha Fidela, Meisye Putri Azizah, & Septia Rizka Hidayah. (2023). Tren Pengembangan Aplikasi Mobile: Sebuah Tinjauan Literatur. *Jurnal Teknik Mesin*,

*Industri, Elektro dan Informatika*, 2(4), 30–48.

<https://doi.org/10.55606/jtmei.v2i4.2848>

Wijaya, M. M., Rohiim, H. Z. N., Ramadani, R. J., & Irawan, R. H. (2025). Pengembangan sistem pengelolaan sampah pintar berbasis IoT untuk efisiensi pengumpulan sampah di kampus. *Prosiding Seminar Nasional Teknologi dan Sains Tahun 2025*, 4, 66–72.

Yas, Q., Alazzawi, A., & Rahmatullah, B. (2023). A comprehensive review of software development life cycle methodologies: Pros, cons, and future directions. *Iraqi Journal for Computer Science and Mathematics*, 173–190.

<https://doi.org/10.52866/ijcsm.2023.04.04.014>

## LAMPIRAN

### a. Proses Wawancara dengan Petugas Kebersihan



### b. Proses Wawancara dengan Mahasiswi



c. Proses Pengujian Alat dengan Petugas Kebersihan



d. Proses Pengujian Alat dengan Mahasiswa



