



# **ECOSPHERA**

Software Requirements

Specification for

EcoSphera Project

26 September 2024

Version 1.0

Prepared By

Kelompok 1

## Riwayat Perubahan

Tanggal	Versi	Keterangan	Diperbaharui Oleh
11 November 2024	0.1	Penambahan Persyaratan Non-fungsional	<ul style="list-style-type: none"> <li>• Zaky Rizzan Zain</li> <li>• Muhammad Raihan Ananditha</li> </ul>
13 Oktober 2024	0.2	Penambahan Flowmap	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan Zain</li> <li>• Salsabila Rahma</li> <li>• Muhammad Raihan</li> </ul>
14 Oktober 2024	0.3	Penambahan DFD Level 0, Level 1 dan Pembuatan Jadwal Schedule	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Muhammad Raihan</li> </ul>
19 Oktober 2024	0.4	Penambahan Project Metodology	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Muhammad Raihan</li> </ul>
19 Oktober 2024	0.5	Penambahan DFD Level 2	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Muhammad Raihan</li> </ul>
20 Oktober 2024	0.6	<ul style="list-style-type: none"> <li>• Penambahan Kamus Data</li> <li>• Project Budget</li> <li>• Project Methodology</li> <li>• Project Schedule</li> </ul>	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Muhammad Raihan</li> </ul>
08 Desember 2024	0.7	<ul style="list-style-type: none"> <li>• Update Flowmap</li> <li>• Desain Arcitecture</li> <li>• User Interface Design</li> <li>• Database Design</li> </ul>	<ul style="list-style-type: none"> <li>• Rafi Islami</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Ramadhania</li> </ul>

			<ul style="list-style-type: none"> <li>• Muhammad Raihan</li> </ul>
03 Januari 2025		<ul style="list-style-type: none"> <li>• Update Architecture Design</li> <li>• Update User Interface Design</li> <li>• Update Database Design</li> <li>• Implementasi</li> <li>• Pengujian</li> </ul>	<ul style="list-style-type: none"> <li>• Rafi Islami Pasha</li> <li>• Zaky Rizzan</li> <li>• Salsabila Rahma</li> <li>• Muhammd Raihan</li> </ul>

### Persetujuan Dokumen

Nama	Peran	Tanda Tangan
Rafi Islami Pasha Dini Hari Putra	Project Manager	

## DAFTAR ISI

<b>DAFTAR ISI .....</b>	<b>iv</b>
<b>1. Pendahuluan.....</b>	<b>1</b>
1.1 Tujuan Pembuatan Dokumen .....	1
1.2 Ruang Lingkup EcoSphera .....	1
1.3 Stakeholder.....	2
1.4 Ringkasan Mengenai EcoSphera .....	3
1.4.1 Katalog Bank Sampah.....	3
1.4.2 Akun Pengelola dan Pengguna.....	3
1.4.3 EcosCalc.....	4
1.4.4 Rating Bank Sampah.....	4
<b>2. Deskripsi Keseluruhan .....</b>	<b>5</b>
2.1 Perspektif Produk .....	5
2.1.1 Sistem Manajemen Bank Sampah.....	5
2.1.2 Tampilan Pengguna .....	5
2.1.3 Tampilan Pengelola Bank Sampah.....	5
2.1.4 Tampilan Pengembangng .....	5
2.2 Fungsi Produk .....	6
2.3 Karakteristik Pengguna .....	6
2.4 Kendala .....	7
2.5 Asumsi dan Ketergantungan .....	8
2.6 Pembagian Persyaratan .....	8
<b>3. Persyaratan Spesifik.....</b>	<b>10</b>
3.1 Persyaratan Fungsional .....	10
3.1.1 Manajemen Akun .....	10
3.1.2 Pencarian Bank Sampah.....	10
3.1.3 Menampilkan Data Bank Sampah.....	11
3.1.4 EcosCalc.....	11
3.2 Persyaratan Non-Fungsional .....	12
3.2.1 Persyaratan Keamanan .....	12
3.2.1 Persyaratan Kegunaan.....	12
3.2.2 Persyaratan Kinerja .....	13

3.2.3	Persyaratan Keandalan .....	13
3.2.4	Persyaratan Kompatibilitas .....	14
3.2.5	Persyaratan Pemeliharaan .....	14
<b>4.</b>	<b>Flowmap .....</b>	<b>16</b>
4.1	EcosCalc.....	16
4.2	Register dan Log In .....	17
4.2.1	Register Pengelola.....	17
4.2.2	Register User.....	18
4.2.3	Log in .....	19
4.3	Search Bank Sampah.....	20
4.4	Create Bank Sampah .....	21
4.5	Edit Data Bank Sampah .....	22
4.6	Delete Bank Sampah .....	23
4.7	Rating Bank Sampah.....	24
<b>5.</b>	<b>Data Flow Diagram.....</b>	<b>25</b>
5.1	Diagram Context.....	25
5.2	Diagram DFD Level 1.....	26
5.3	Diagram DFD Level 2.....	28
5.3.1	Diagram DFD Level 2 Register .....	28
5.3.2	Diagram DFD Level 2 Login .....	29
5.3.3	Diagram DFD Level 2 CRUD Bank Sampah .....	30
5.3.4	Diagram DFD Level 2 Search Bank Sampah .....	31
5.3.5	Diagram DFD Level 2 EcosCalc .....	31
5.3.6	Diagram DFD Level 2 Rating Bank sampah .....	32
<b>6.</b>	<b>Project Methodology.....</b>	<b>34</b>
<b>7.</b>	<b>Project Schedule.....</b>	<b>35</b>
7.1	Analisis Kebutuhan .....	36
7.2	Pembuatan Flowmap .....	36
7.3	Pembuatan DFD (Data Flow Diagram).....	36
7.4	Update SKPL .....	37
7.5	Pembuatan Design EcoSphera .....	37
7.6	Implementasi Kodingan .....	37
<b>8.</b>	<b>Project Budget.....</b>	<b>38</b>
<b>9.</b>	<b>Software Design.....</b>	<b>38</b>

9.1	Architecture Design .....	38
9.2	User Interface Design.....	41
9.2.1	Design System.....	41
9.2.2	Wireframe.....	42
a.	Halaman Registrasi Pengelola.....	42
d.	Halaman Dashboard Pengguna .....	44
9.2	Database Design.....	52
<b>10.</b>	<b>Implementasi .....</b>	<b>54</b>
<b>11.</b>	<b>Pengujian .....</b>	<b>66</b>
11.1	Lingkungan Pengujian .....	66
11.1.1	Perangkat Lunak Pengujian.....	66
11.1.2	Perangkat Keras Pengujian .....	66
11.1.3	Material Pengujian .....	66
11.1.4	Sumber Daya Manusia .....	66
11.1.5	Prosedur Umum Pengujian .....	66
11.2	Hasil Pengujian .....	68
<b>Lampiran</b>	.....	<b>88</b>

## **1. Pendahuluan**

### **1.1 Tujuan Pembuatan Dokumen**

Tujuan dari dokumen ini adalah untuk menyediakan serangkaian persyaratan yang komprehensif untuk proyek EcoSphera. Yang akan menguraikan persyaratan fungsional dan non-fungsional untuk sistem perangkat lunak, serta kendala dan asumsi yang memengaruhi proses pengembangan.

Dokumen ini berfungsi sebagai alat komunikasi antara tim pengembang dan pemangku kepentingan lain dalam proyek. Dan akan memberikan pemahaman yang jelas tentang persyaratan untuk sistem perangkat lunak, sehingga memungkinkan para pemangku kepentingan untuk dapat membuat Keputusan yang tepat tentang desain, pengembangan, pengujian, dan penyebaran sistem.

Persyaratan yang diuraikan dalam dokumen ini dimaksudkan untuk memandu tim pengembangan dalam pembuatan sistem perangkat lunak berkualitas tinggi yang memenuhi kebutuhan pengguna EcoSphera. Dokumen ini juga berfungsi sebagai dasar untuk pengujian dan verifikasi, memastikan bahwa sistem perangkat lunak memenuhi fungsionalitas dan standar kualitas yang diinginkan.

Singkatnya, tujuan dari dokumen ini adalah untuk memberikan serangkaian persyaratan yang jelas dan komprehensif untuk proyek EchoSphera, memungkinkan pemangku kepentingan untuk membuat keputusan yang tepat tentang pengembangan dan penyebaran sistem perangkat lunak.

### **1.2 Ruang Lingkup EcoSphera**

EcoSphera merupakan website yang diciptakan untuk memfasilitasi pengelolaan sampah sebagai bentuk kemudahan dan peningkatan kesadaran masyarakat akan pentingnya menjaga kebersihan lingkungan. Ruang lingkup dari proyek ini terbatas untuk memberikan informasi mengenai bank sampah yang ada di sekitar user mengenai pengelola, lokasi, jadwal operasi, informasi apakah ada currency pada bank sampah di sekitar Kampus UPI Cibiru.

EcoSphera dibuat dikarenakan adanya permasalahan lingkungan akibat kurangnya kesadaran dari masyarakat akan pentingnya pengelolaan sampah yang baik. Hal ini menyebabkan terjadinya penumpukan sampah dan berdampak pada kerusakan lingkungan.

Produk EcoSphera akan mempunyai fitur utama diantaranya:

- a. Penelusuran dan pencarian bank sampah: Pengguna dapat menelusuri dan mencari bank sampah berdasarkan lokasi, dan melihat indormasi terperinci, termasuk informasi pengelola dan sistem pertukaran yang ada pada masing-masing bank sampah.
- b. Mendaftarkan dan mencantumkan informasi bank sampah: Pengelola bank sampah dapat mendaftarkan bank sampah yang dikelolanya serta memberikan informasi mengenai profile pengelola, jam operasional, lokasi, dan sistem pertukaran.
- c. EcosCalc: Pengguna dapat menghitung harga tukar sampah berdasarkan jumlah dan jenis sampah yang akan ditukarkan.

### 1.3 Stakeholder

Beberapa stakeholder yang terlibat dalam pengembangan dan pengimplementasian sistem perangkat lunak EcoSphera:

- a. Tim Pengembang

Tim pengembang akan bertanggung jawab atas desain, pengembangan, dan implementasi sistem perangkat lunak. Mereka akan bekerja sama untuk memastikan bahwa sistem perangkat lunak memenuhi semua persyaratan dan spesifikasi.

- b. Pengelola Bank Sampah

Peran Pengelola bank sampah adalah mengelola bank sampah dan mengelola akun bank sampahnya. Ia juga bertanggung jawab untuk mencantumkan informasi terkait operasional bank sampah dan layanan bank sampahnya kepada pengguna.

- c. Dosen

Dosen akan memantau progress dari tim pengembang dan akan melakukan penilaian untuk produk yang dikembangkan.

d. Pengguna

Pengguna akan menggunakan sistem perangkat lunak untuk mencari bank sampah, menemukan informasi tentang bank sampah, menemukan informasi tentang pengelola bank sampah dan mengelola akun pengguna mereka. Sistem perangkat lunak harus ramah pengguna dan memberikan pengalaman pengguna yang positif untuk memastikan kepuasan pelanggan.

#### **1.4 Ringkasan Mengenai EcoSphera**

Ecosphera adalah platform online yang menawarkan pengguna untuk menjadikan bank sampah mereka menjadi digital dengan mendaftarkan bank sampah ke website Ecosphera. Setelah pengguna mendaftarkan bank sampah pengguna ke website ecosphera, pengguna dapat menentukan harga sampah sesuai dengan yang pengguna inginkan. Kemudian, pelanggan dari bank sampah yang telah terdaftar di website ecosphere dapat mengakses bank sampah tersebut secara digital. Pelanggan dapat melihat harga sampah disetiap bank sampah melalui website ecosphere dan dapat melihat Lokasi bank sampah. Pelanggan juga dapat memilih bank sampah sesuai dengan kecocokan pelanggan.

EcoSphera akan terdiri dari fitur-fitur utama berikut:

##### **1.4.1 Katalog Bank Sampah**

Katalog Bank Sampah akan memperlihatkan list bank sampah yang terdaftar di website EcoSphera. Ketika pelanggan menekan salah satu bank sampah, maka akan terlihat deskripsi, informasi mengenai currency setiap jenis sampah, jam operasional, rating dan lokasi bank sampah.

##### **1.4.2 Akun Pengelola dan Pengguna**

- a. Pengelola bank sampah dapat mendaftarkan bank sampah mereka dengan menekan opsi “daftarkan bank sampah”. Setelah itu pengelola diharuskan mengisi profile bank sampah mereka lalu setelah semua proses pendaftaran selesai, bank sampah akan muncul di tampilan pengguna dan pengelola dapat login untuk mengakses akun mereka.
- b. Pengguna website EcoSphera dapat mendaftarkan diri mereka dengan menekan opsi “daftar sekarang”. Setelah itu pengguna akan diminta untuk

mengisi profile mereka. Setelah proses pendaftaran selesai, pengguna dapat login untuk mengakses bank sampah.

#### **1.4.3 EcoCalc**

EcoScalc adalah fitur kalkulasi untuk menghitung nilai tukar dari sampah yang akan ditukarkan. Setelah pengguna memilih bank yang mereka inginkan, pengguna juga dapat menginput jenis sampah beserta beratnya kemudian secara otomatis akan terlihat harga tukar dari sampah tersebut sesuai dengan yang ditetapkan oleh pengelola. Sehingga pelanggan dapat melakukan perbandingan dengan bank sampah lainnya.

#### **1.4.4 Rating Bank Sampah**

Pengguna dapat melakukan rating kepada bank sampah sesuai dengan pengalaman pengguna terkait bank sampah, bank sampah dengan rating tertinggi akan ditampilkan paling atas dibanding bank sampah lainnya.

## **2. Deskripsi Keseluruhan**

### **2.1 Perspektif Produk**

EcoSphera akan menjadi website yang dirancang dan dikembangkan untuk memberikan informasi mengenai bank sampah. Sistem perangkat lunak akan dikembangkan dengan menggunakan teknologi pengembangan web modern.

Sistem perangkat lunak akan berjalan bersandingan dengan berbagai sistem eksternal untuk mendukung kenyamanan pengguna. Sistem eksternal berikut akan diintegrasikan dengan EcoSphera;

#### **2.1.1 Sistem Manajemen Bank Sampah**

Ecosphera akan terintegrasi dengan sistem manajemen bank sampah yang berfungsi untuk menerima informasi terbaru dari pengelola bank sampah. Sistem ini akan bertanggung jawab untuk menyimpan dan mempertahankan informasi terkini dari bank sampah tentang jam operasional ataupun aturan terbaru dari pengelola bank sampah.

#### **2.1.2 Tampilan Pengguna**

Sistem ini akan bertugas untuk membuat akun pengguna dapat mencari bank sampah yang di inginkan dan mendapatkan informasi detail tentang bank sampah tersebut. Pengguna juga dapat mendapatkan lokasi detail dari bank sampah dan dapat melakukan kalkulasi terhadap nilai sampah yang akan ditukar.

#### **2.1.3 Tampilan Pengelola Bank Sampah**

Sistem ini akan bertugas untuk membuat akun pengelola dapat mencantumkan informasi dan aturan dari bank sampah yang dikelolanya.

#### **2.1.4 Tampilan Pengembang**

Sistem akan bertugas untuk membuat pengembang dapat melakukan pemeliharaan terhadap website yang telah dibuat.

## **2.2 Fungsi Produk**

EchoSphera akan menyediakan beberapa fungsi untuk pengguna:

a. Manajemen Akun Pengguna dan Pengelola

Pengguna dan pengelola dapat membuat dan mengelola akun seperti menyimpan informasi pribadi dan melakukan update terhadapnya.

b. Penelusuran dan Pencarian Bank Sampah

Pengguna dapat melakukan pencarian bank sampah yang sudah terdaftar pada website melalui menu Searchbox.

c. Manajemen Bank Sampah

Pengelola bank sampah dapat mengelola informasi mengenai bank sampah yang dikelolanya, termasuk informasi identitas pengelola, jam operasional, dan aturan yang ada pada bank sampah.

d. Tampilan Bank Sampah

Pengguna dapat melihat informasi yang telah dicantumkan oleh pengelola.

e. Sistem Kalkulasi

Pengguna dapat mengkalkulasikan nilai tukar dari sampah yang akan ditukarkan berdasarkan jenis dan berat sampah. Sistem ini akan ada pada setiap bank sampah yang mencantumkan sistem tukar pada bank sampahnya.

f. Sistem Rating

Pengguna dapat melakukan penilaian terhadap pelayanan yang diberikan bank sampah. Bank sampah dengan rating tertinggi akan berada diurutan teratas.

g. Customer Support

Pengguna dapat melaporkan masalah atau kesulitan yang dialami di dalam platform EcoSphera dan akan mendapatkan Solusi dengan cepat dan efisien, yang kedepannya akan dapat meningkatkan loyalitas dan kepuasan pelanggan.

## **2.3 Karakteristik Pengguna**

EcoSphera dirancang untuk melayani berbagai pengguna, diantaranya ialah pengguna yang tertarik untuk menelusuri dan dantang ke bank sampah. Pengguna ini mungkin memiliki kesadaran akan pentingnya menjaga lingkungan.

Berikut adalah beberapa karakteristik pengguna utama yang telah dipertimbangkan dalam pengembangan EcoSphera:

a. Memiliki kemahiran

Pengguna mungkin memiliki kemahiran dan pengalaman dalam penggunaan suatu website. Sistem EcoSphera dirancang agar ramah pengguna dengan dilengkapi sistem navigasi dan fitur yang mudah digunakan.

b. Bahasa

Untuk memudahkan masyarakat dalam penggunaannya EcoSphera hanya akan tersedia dalam Bahasa Indonesia.

c. Aksebilitas

Website dapat diakses oleh seluruh masyarakat.

d. Keamanan

Website mengharuskan pengguna untuk membuat akun dan masuk untuk mengakses fitur-fitur yang ada pada EcoSphere.

Kesimpulannya, EcoSphera dirancang untuk melayani berbagai pengguna termasuk pengelola bank sampah, dan masyarakat. EcoSphera dirancang dengan mempertimbangkan berbagai karakteristik pengguna.

## 2.4 Kendala

Pengembangan proyek EcoSphera tentunya memiliki kendala yang perlu dipertimbangkan, diantaranya:

a. Keterbatasan Ilmu

Pengembangan proyek EcoSphera membutuhkan pengetahuan dan pengalaman dalam pembuatan dan pengelolaan website. Sedangkan para pengembang masih memiliki keterbatasan ilmu dan pengalaman karena tidak pernah terlibat dalam pembuatan website.

b. Keterbatasan Teknologi

Proyek EcoSphera ini terbatas hanya dengan menggunakan bahasa pemrograman python dan tidak diperkenankan menggunakan framework.

c. Keterbatasan Relasi

Untuk kelancaran proyek ini, pengembang perlu bekerjasama dengan para pengelola bank sampah. Hal itu menjadi kendala karena kurangnya informasi mengenai informasi mengenai pengelola bank sampah.

- d. Kurangnya Informasi Mengenai Bank Sampah.

Kurangnya informasi mengenai cara kerja pengelolaan sampah melalui bank sampah dan lokasi dimana adanya bank sampah menjadi kendala karena dibutuhkan analisis lebih lanjut.

- e. Keterbatasan Waktu Penggerjaan

Pengembangan proyek EcoSphera memiliki tenggat waktu yang cukup singkat. Sehingga dibutuhkan perencanaan dan pelaksanaan yang cepat dan tepat. Dibutuhkan juga komunikasi tim yang baik untuk memastikan proyek selesai tepat waktu.

## **2.5 Asumsi dan Ketergantungan**

Pengembangan EcoSphera didasarkan pada sejumlah asumsi dan *dependencies* yang penting untuk dipertimbangkan selama proses desain dan implementasi. Diantaranya adalah:

### **Asumsi:**

- a. EcoSphera dapat dikembangkan secara daring oleh seluruh tim pengembang.
- b. EcoSphera menggunakan sistem bank sampah yang sudah banyak diterapkan dan tidak perlu membuat sistem baru yang belum diketahui banyak orang.
- c. Pengembangan EcoSphera tahap 1 direncanakan akan selesai pada akhir semester 1 pada tahun 2024.

### **Ketergantungan:**

- a. EcoSphera sangat bergantung pada sistem bank sampah yang sudah ada dan harus bisa bekerja sama dengan pengelola bank sampah yang ada.
- b. Pengembangan EcoSphera bergantung pada dokumentasi teknis dari bahasa yang digunakan.

Kesimpulannya, sangat penting untuk mempertimbangkan asumsi dan *dependencies* selama proses pengembangan untuk memastikan produk EcoSphera selesai tepat waktu dan sesuai dengan semua persyaratan.

## **2.6 Pembagian Persyaratan**

Persyaratan untuk EcoSphera dibagi kedalam beberapa fase yang didasarkan pada kekritisan dan kompleksitas setiap persyaratan. Pembagian persyaratan adalah sebagai berikut:

**Fase 1: Persyaratan Dasar**

- Manajemen akun
- Pencarian bank sampah
- Menampilkan data bank sampah

**Fase 2: Persyaratan Lanjutan**

- Menghitung harga tukar sampah berdasarkan jenis dan jumlah yang ditentukan pengguna.
- Rating bank sampah.

### **3. Persyaratan Spesifik**

#### **3.1 Persyaratan Fungsional**

##### **3.1.1 Manajemen Akun**

Bagian ini menguraikan persyaratan fungsional yang terkait dengan manajemen akun pengguna dan pengelola pada EcoSphera.

###### a. Registrasi Pengguna

Pengguna dapat mendaftarkan diri kedalam platform EcoSphera dengan cara membuat akun menggunakan username dan password.

###### b. Login Pengguna

Pengguna dapat masuk ke akun pribadi nya masing-masing yang telah dibuat sebelumnya pada tahap registrasi akun.

###### c. Registrasi Pengelola

Pengelola dapat mendaftarkan bank sampah yang dikelolanya kedalam platform EcoSphera dengan cara membuat akun sebagai pengelola bank sampah.

###### d. Login Pengelola

Pengelola yang telah membuat akun pada tahap registrasi pengelola, dapat langsung masuk ke akun pengelola bank sampahnya.

###### e. Rating pengguna untuk bank sampah

Pengguna dapat memberikan rating atau penilaian terhadap bank sampah.

###### f. Rating untuk bank sampah

Pengelola bank sampah, dapat menerima rating atau penilaian dari pengguna untuk bank sampahnya.

##### **3.1.2 Pencarian Bank Sampah**

EcoSphera harus menyediakan tampilan yang menarik dan mudah digunakan agar memudahkan pengguna dalam menggunakan dan mencari bank sampah yang sesuai dengan kriteria pengguna. Persyaratan yang harus dipenuhi diantaranya:

###### a. Mesin pencari dalam EcoSphera harus bisa mencari bank sampah berdasarkan lokasi, pengelola bank sampah, nama bank sampah.

- b. Mesin pencari dalam EcoSphera harus bisa mengurutkan hasil pencarian berdasarkan rating yang dimiliki oleh bank sampah.
- c. Mesin pencari dalam EcoSphera harus bisa menampilkan hasil yang sesuai dengan apa yang dicari oleh pengguna.
- d. Hasil pencarian yang dikeluarkan harus bisa mengalihkan pengguna ke halaman yang dituju, jika salah satu dari hasil pencarian tersebut di klik.

### **3.1.3 Menampilkan Data Bank Sampah**

EcoSphera harus menampilkan data bank sampah yang terdaftar dalam platform EcoSphera, dan ditampilkan berdasarkan rating terbesar sampai terkecil.

- a. Memasukkan data bank sampah

Pengelola dapat memasukkan data bank sampah kedalam EcoSphera agar bank sampahnya dapat dilihat oleh para pengguna EcoSphera.

- b. Edit data bank sampah

Pengelola juga dapat mengedit data bank sampah yang dikelolanya.

- c. Hapus data bank sampah

Pengelola dapat menghapus data bank sampah yang dikelola olehnya.

- d. Menampilkan data bank sampah

Data bank sampah akan ditampilkan pada platform EcoSphera baik dihalaman Pengguna, maupun Pengelola.

### **3.1.4 EcosCalc**

EcoSphera harus menyediakan fitur untuk menghitung berapa uang yang akan diterima oleh pengguna sesuai dengan jenis dan jumlah sampah yang akan ditukarkan melalui EcosCalc. Persyaratan yang harus dipenuhi:

- a. EcosCalc harus bisa menghitung harga dalam satuan Kg.
- b. EcosCalc harus bisa menentukan harga sesuai dengan jenis sampah yang dipilih pengguna.
- c. EcosCalc harus bisa menghitung uang yang akan didapatkan oleh pengguna lebih dari satu jenis sampah sekaligus.

## **3.2 Persyaratan Non-Fungsional**

Persyaratan non-fungsional menjelaskan bagaimana sistem seharusnya berperilaku, bukan apa yang seharusnya dilakukan. Persyaratan ini biasanya melibatkan faktor-faktor seperti kinerja, keandalan, skalabilitas, dan keamanan. Berikut ini adalah beberapa contoh persyaratan non-fungsional pada EcoSphera:

### **3.2.1 Persyaratan Keamanan**

Persyaratan keamanan menjelaskan bagaimana sistem harus melindungi informasi sensitive, mencegah akses tidak sah, dan memastikan integritas dan kerahasiaan data. Berikut adalah persyaratan keamanan dalam EcoSphera:

- a. Autentikasi: Sistem harus mengharuskan pengguna untuk mengautentikasi diri mereka sendiri sebelum mengakses informasi sensitif apa pun, menggunakan kebijakan kata sandi yang kuat untuk keamanan tambahan.
- b. Otorisasi: Sistem harus menerapkan kontrol akses berbasis peran, memastikan bahwa pengguna hanya dapat mengakses informasi dan fungsi yang mereka berwenang untuk menggunakannya.

### **3.2.1 Persyaratan Kegunaan**

Persyaratan kegunaan menjelaskan seberapa mudah dan ramah pengguna sistem tersebut. Berikut ini adalah persyaratan kegunaan dalam EcoSphera:

- a. Fungsionalitas mesin pencari: Sistem harus memiliki fungsi pencarian yang kuat dan akurat yang memungkinkan pengguna menemukan produk dan informasi dengan cepat.
- b. Tampilan responsif: Sistem harus responsif terhadap perangkat seluler, dengan tata letak dan desain yang beradaptasi dengan berbagai ukuran dan resolusi layar.
- c. Pesan kesalahan: Sistem harus menampilkan pesan kesalahan yang jelas dan membantu saat terjadi kesalahan, disertai petunjuk tentang cara memperbaiki masalah tersebut.
- d. Konsisten: Sistem harus konsisten dalam hal tata letak, desain, dan terminologi, dengan tampilan dan nuansa yang konsisten di semua halaman dan bagian.

### **3.2.2 Persyaratan Kinerja**

Persyaratan kinerja menjelaskan bagaimana sistem seharusnya berfungsi dalam hal efisiensi operasional. Persyaratan ini bertujuan untuk memastikan bahwa aplikasi atau sistem perangkat lunak dapat memberikan respons yang cepat, stabil, dan efisien ketika digunakan oleh pengguna. Berikut ini adalah persyaratan kinerja dalam EcoSphera:

- a. Waktu respons: Sistem harus mampu memberikan respons terhadap permintaan pengguna (seperti pencarian bank sampah dan kalkulasi nilai tukar) dalam waktu kurang dari 2 detik. Keterlambatan yang lebih dari 3 detik dapat menyebabkan pengalaman pengguna yang buruk.
- b. Kapasitas server: Sistem harus mampu menerima permintaan minimal 1000 permintaan dari pengguna aktif tanpa mengalami penurunan kinerja signifikan, dengan waktu respons yang tetap optimal.

### **3.2.3 Persyaratan Keandalan**

Persyaratan keandalan menjelaskan menjelaskan sejauh mana sistem dapat beroperasi dengan baik dan konsisten sesuai yang diharapkan dalam jangka waktu tertentu, tanpa terjadi gangguan yang signifikan. Keandalan mengukur kemampuan sistem untuk memberikan layanan yang stabil dan berkelanjutan kepada pengguna. Berikut ini adalah persyaratan keandalan dalam EcoSphera:

- a. Waktu aktif (*Uptime*): Sistem harus dapat mencapai uptime setidaknya 99.99%, artinya layanan harus selalu tersedia sepanjang waktu kecuali pada saat pemeliharaan terjadwal. Ini berarti sistem dapat mati tidak lebih dari 52 menit dan 35 detik per tahun.
- b. Redudansi data: Sistem harus memiliki server cadangan yang siap digunakan apabila server utama mengalami kegagalan. Sehingga, layanan tidak terganggu.
- c. Jam operasional tidak terbatas: Sistem harus dapat diakses 24x7 kecuali saat jadwal pemeliharaan sistem.

### **3.2.4 Persyaratan Kompatibilitas**

Persyaratan kompatibilitas menjelaskan tentang persyaratan spesifikasi minimal yang dibutuhkan untuk menjalankan sistem.

- a. Kompatibilitas Sistem Operasi: EcoSphera akan kompatibel dengan sistem operasi:
  - Windows 10 atau yang terbaru
- b. Kompabilitas *Browser*: EcoSphera akan kompatibel dengan *browser* berikut:
  - Chrome (Versi Terbaru)
  - Mozilla Firefox (Versi Terbaru)
  - Microsoft Edge (Versi Terbaru)
- c. Kompabilitas Perangkat Seluler: EcoSphera akan kompatibel dengan perangkat seluler:
  - Android 9 atau versi terbaru
  - Perangkat dengan layer minimal 4 inch
- d. Kompabilitas Basis Data: EcoSphera akan kompatibel dengan basis data:
  - MySQL 8 atau versi terbaru
  - PostgreSQL 12 atau versi terbaru
  - MongoDB 4 atau versi terbaru

Catatan: Persyaratan kompatibilitas di atas dapat berubah berdasarkan pembaruan atau perubahan pada sistem operasi, browser yang disebutkan.

### **3.2.5 Persyaratan Pemeliharaan**

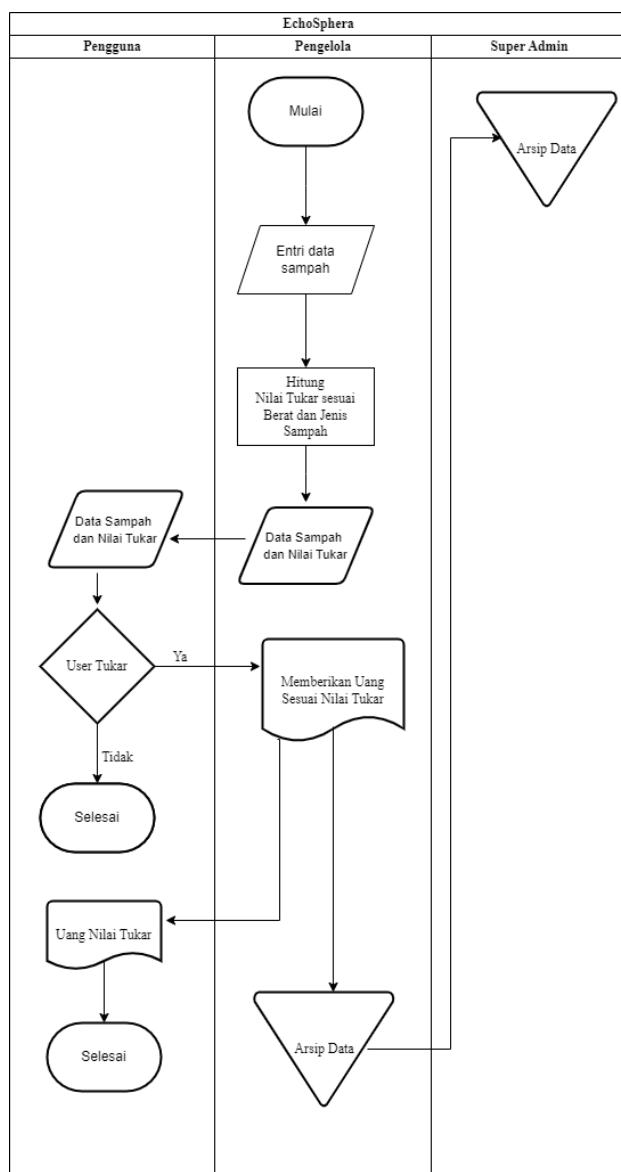
Persyaratan Pemeliharaan menjelaskan kemampuan sistem untuk diperbaiki, ditingkatkan, diperbarui, atau dimodifikasi dengan mudah setelah implementasi awal. Persyaratan pemeliharaan bertujuan untuk memastikan bahwa sistem dapat dengan mudah dikelola dan dipelihara sepanjang siklus hidupnya, baik dalam memperbaiki bug, menambahkan fitur baru, atau meningkatkan kinerja. Berikut ini adalah persyaratan pemeliharaan dalam EcoSphera:

- a. Modularitas sistem: Sistem harus dirancang secara modular, di mana fungsionalitas dibagi ke dalam komponen atau modul yang terpisah. Sehingga jika terdapat kerusakan pada salah satu modul, dalam perbaikannya tidak perlu menonaktifkan modul yang lain.
- b. Dokumentasi lengkap: Sistem harus disertai dokumentasi yang jelas dan lengkap agar mudah dipelihara oleh pengembang di masa yang akan datang.

## 4. Flowmap

### 4.1 EcosCalc

- a. Untuk menggunakan EcosCalc, dimulai dengan memasukkan data sampah ke dalam system
- b. Dari data sampah tersebut akan dihitung nilai tukar sesuai berat dan jenis sampah
- c. Selanjutnya data sampah akan diberikan kepada pengguna untuk menentukan apakah sampah tersebut akan ditukar atau tidak.
- d. Apabila pengguna setuju, pengelola akan memberikan uang sesuai nilai tukar. Apabila tidak setuju, proses pun selesai dilakukan
- e. Data tersebut disimpan oleh pengelola kemudian superadmin.

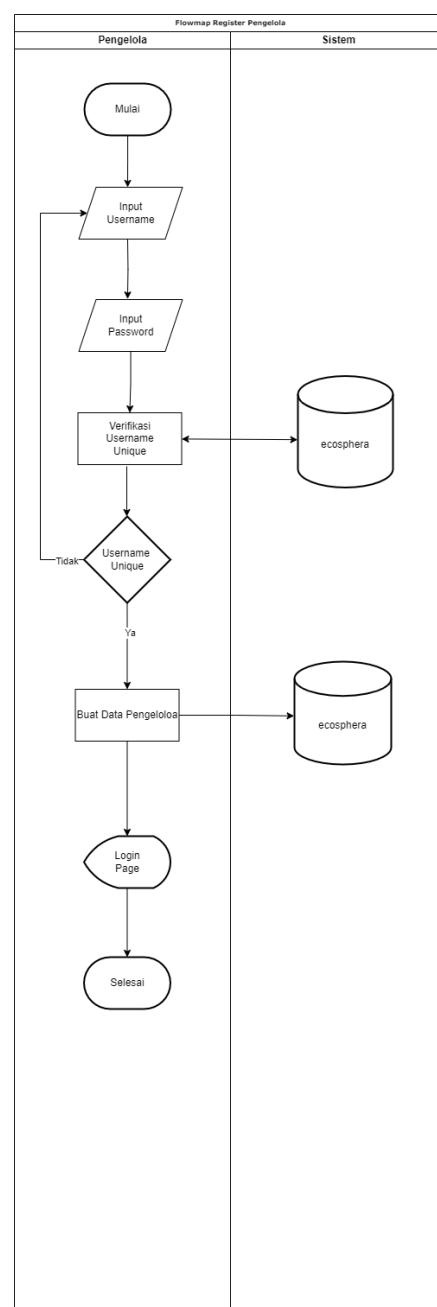


Gambar 4.1 Flowmap EcosCalc

## 4.2 Register dan Log In

### 4.2.1 Register Pengelola

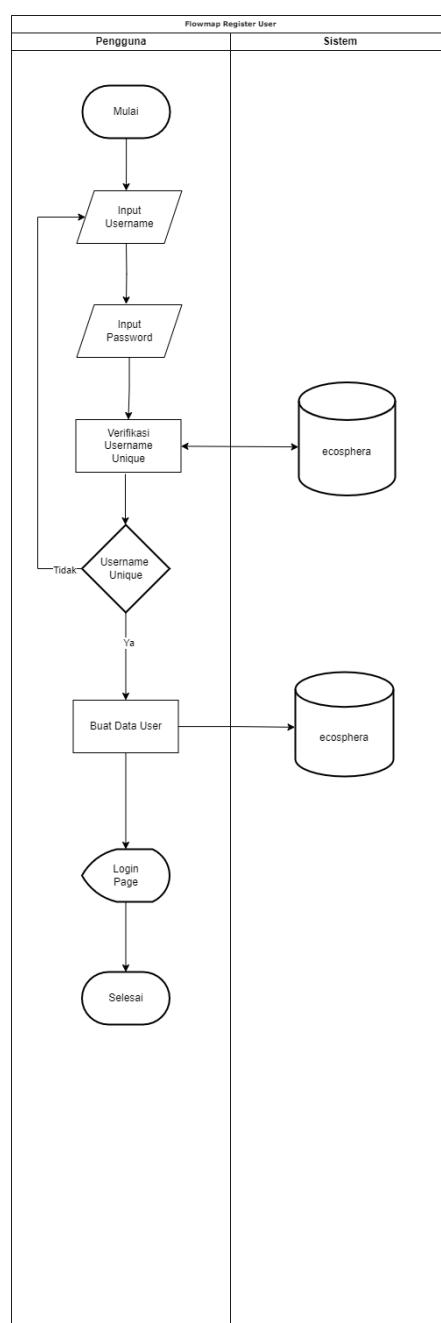
- a. Dimulai dengan pengelola diarahkan untuk menginput username dan password.
- b. Kemudian dilakukan verifikasi username unique. Jika “Ya”, akan dilakukan proses pendataan untuk akun pengelola,
- c. Setelah berhasil, pengelola diarahkan ke halaman log in untuk melakukan proses log in.



Gambar 4.2.1 Flowmap Register Pengelola

#### 4.2.2 Register User

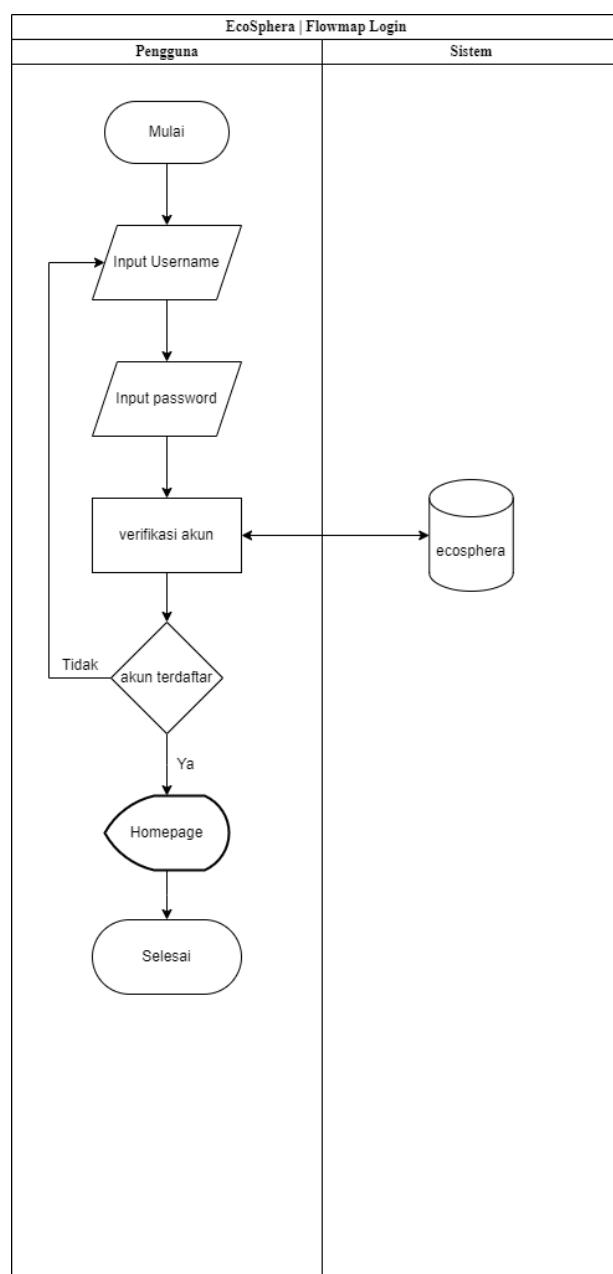
- a. Ketika pertama kali membuka EcoSphera, pengguna akan diarahkan untuk memulai registrasi.
- b. Pengguna diperintahkan menginput username dan password untuk akun EchoSphera.
- c. Kemudian dilakukan verifikasi username unique. Jika “Ya”, akan dilakukan proses pendataan untuk akun pengguna,
- d. Setelah berhasil, pengguna diarahkan ke halaman log in untuk melakukan proses log in.



Gambar 14.2.2 Flowmap Register user

#### 4.2.3 Log in

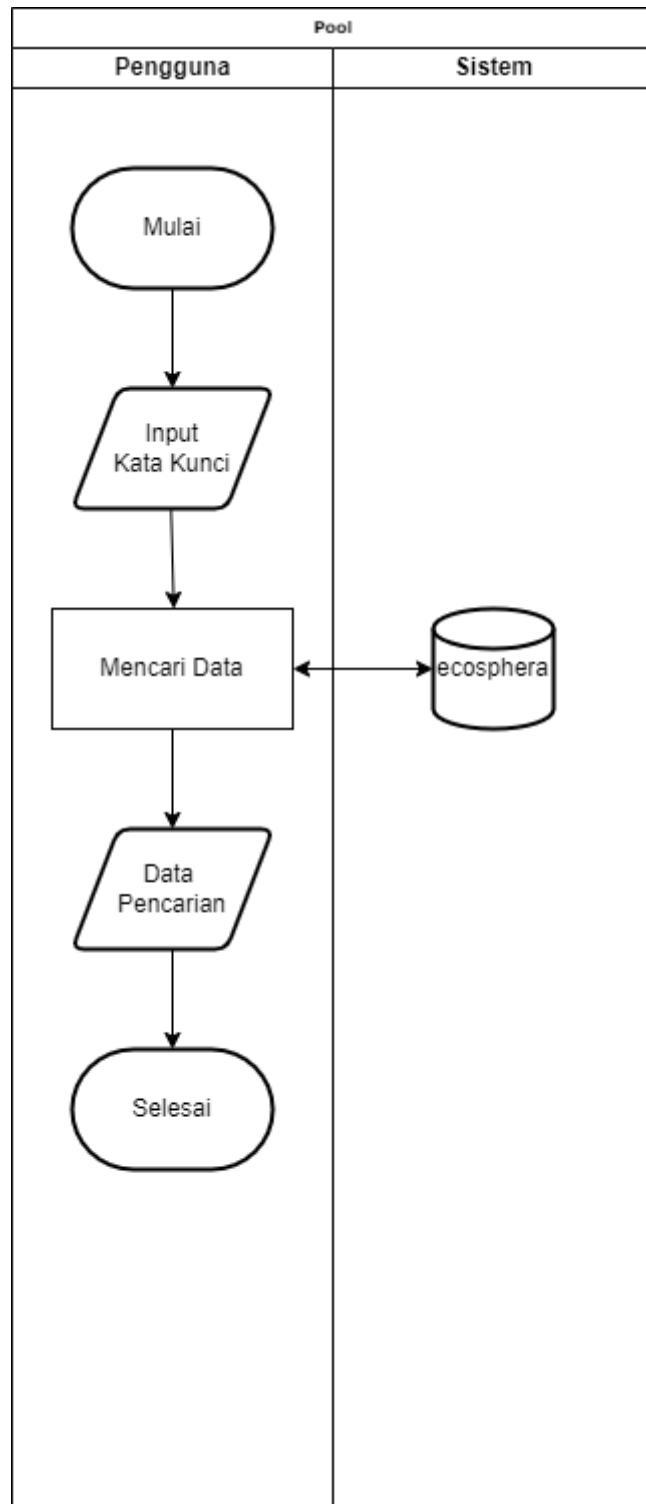
- a. Pengguna diarahkan ke halaman log in.
- b. Dilanjutkan dengan menginput username dan password.
- c. Setelah input username dan password, dilakukan proses verifikasi akun. Apabila “Ya”, yaitu akun berhasil diverifikasi terdaftar, pengguna bisa melanjutkan ke tahap selanjutnya dan apabila “Tidak” pengguna akan diarahkan kembali untuk menginput username dan password Kembali.
- d. Setelah akun berhasil didaftarkan, pengguna akan diarahkan pada menu homepage.



Gambar 4.2.3 Flowmap Log In

### 4.3 Search Bank Sampah

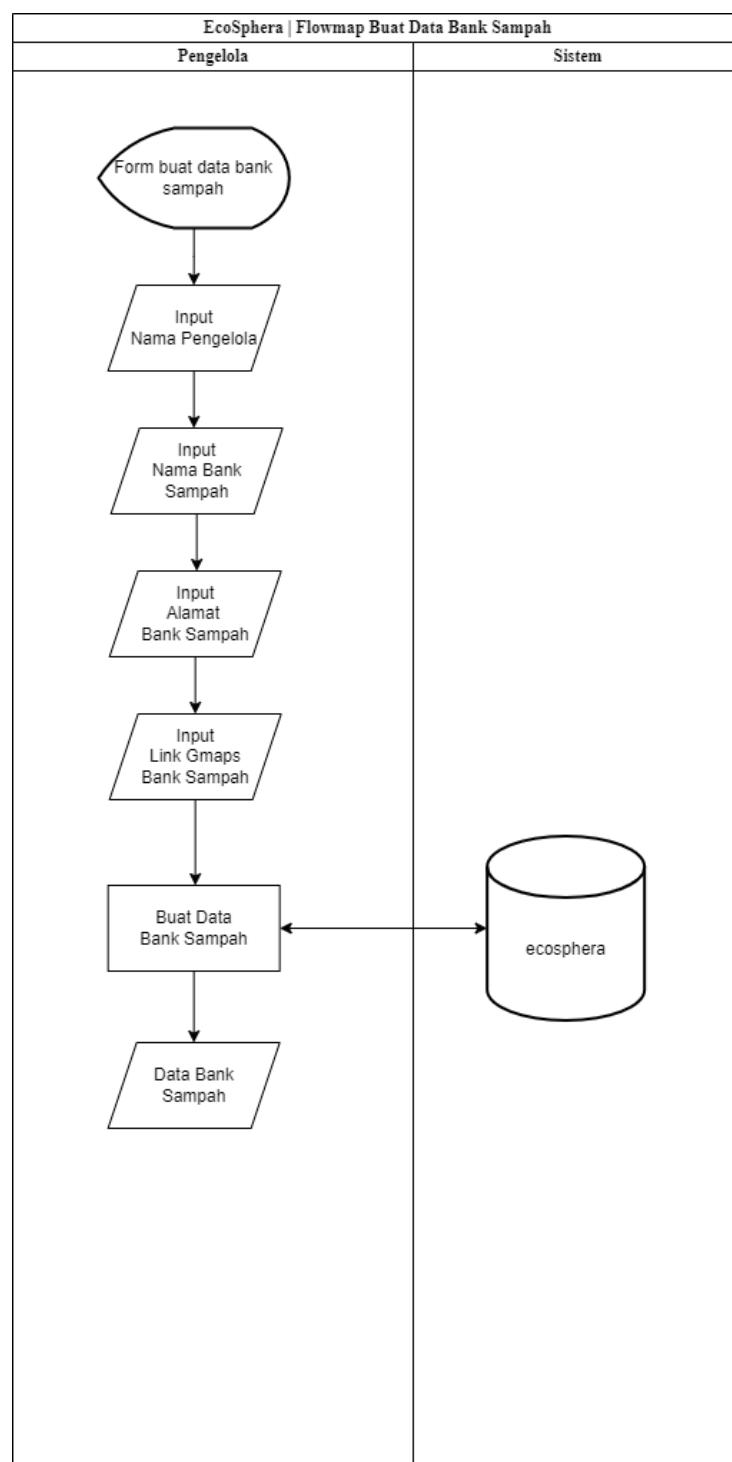
- a. Untuk menggunakan fitur ini, pengguna memasukkan kata kunci.
- b. Selanjutnya dilakukan proses pencarian data.
- c. Setelah berhasil, akan ditampilkan data hasil pencarian.



Gambar 24.3 Search Bank Sampah

#### 4.4 Create Bank Sampah

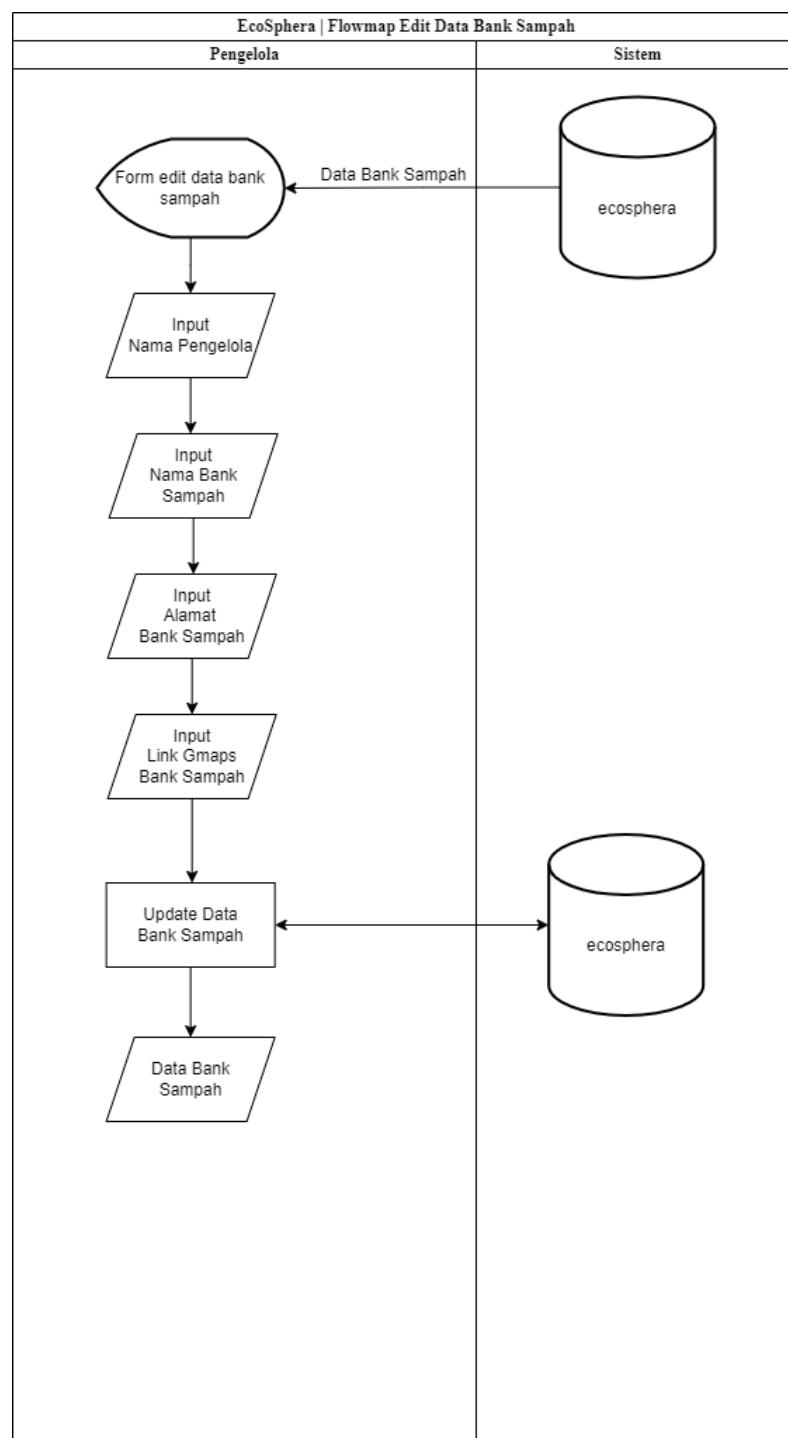
- a. Untuk create bank sampah, pengelola disediakan sebuah form
- b. Pengelola memasukkan nama pengelola, nama bank sampah, alamat bank sampah, dan link gmaps bank sampah.
- c. Selanjutnya data akan diproses
- d. Setelah berhasil, data akan ditampilkan.



Gambar 3.6 Flowmap Create Bank Sampah

#### 4.5 Edit Data Bank Sampah

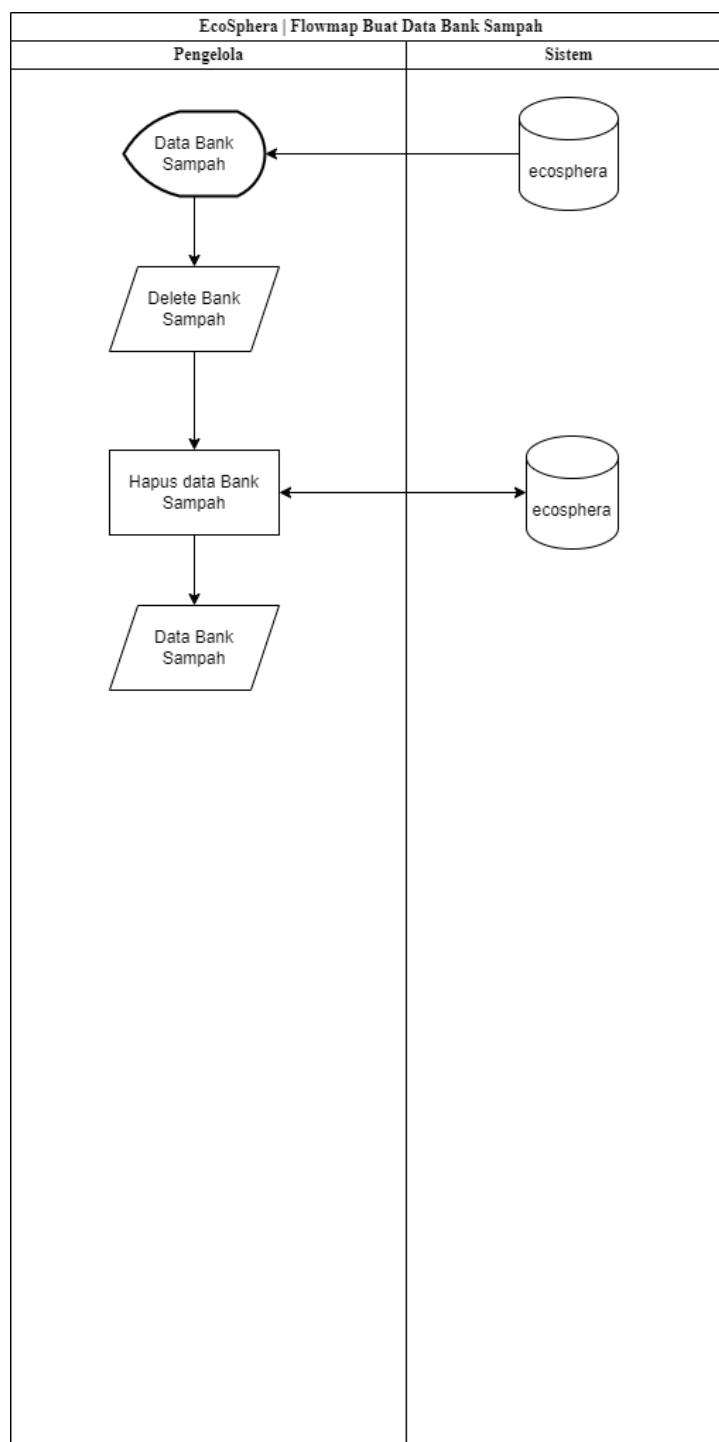
- b. Disediakan form agar pengelola dapat melakukan edit data bank sampah.
- c. Pengelola diperintahkan untuk menginput nama pengelola, nama pengelola, dan link gmaps bank sampah.
- d. Selesai menginput data, akan diproses update data bank sampah.
- e. Setelahnya, ditampilkan data terbaru bank sampah.



Gambar 44.5 Flowmap Edit Data Bank Sampah

#### 4.6 Delete Bank Sampah

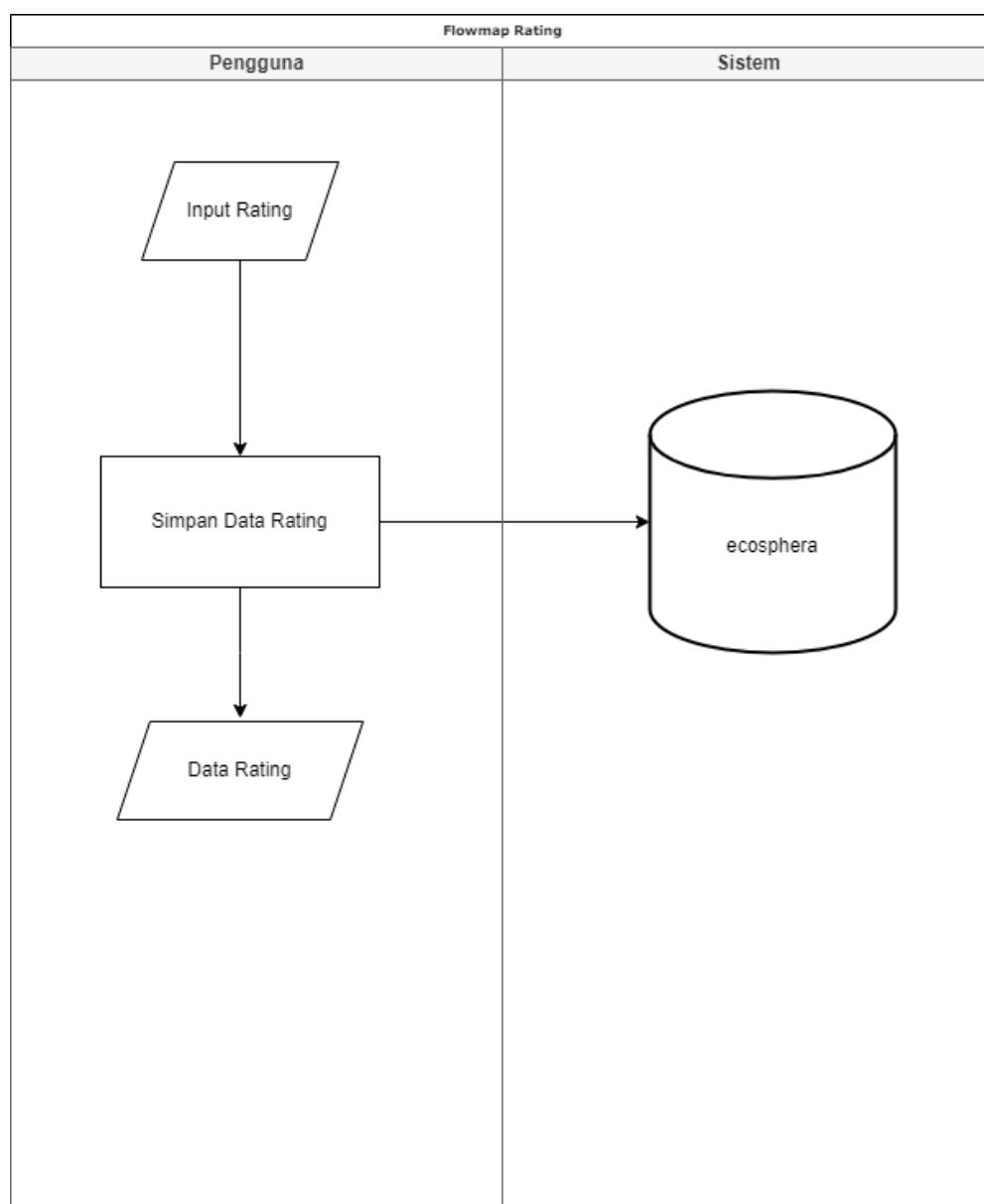
- a. Pengelola dapat menggunakan fitur untuk menghapus data bank sampah dimulai dengan ditampilkan terlebih dahulu data bank sampah.
- b. Setelah data ditampilkan pilih “Delete” untuk melakukan proses delete
- c. Dilakukan proses penghapusan data bank sampah
- d. Kemudian akan dilakukan data tebaru.



Gambar 4.4 Flowmap Delete Bank Sampah

#### 4.7 Rating Bank Sampah

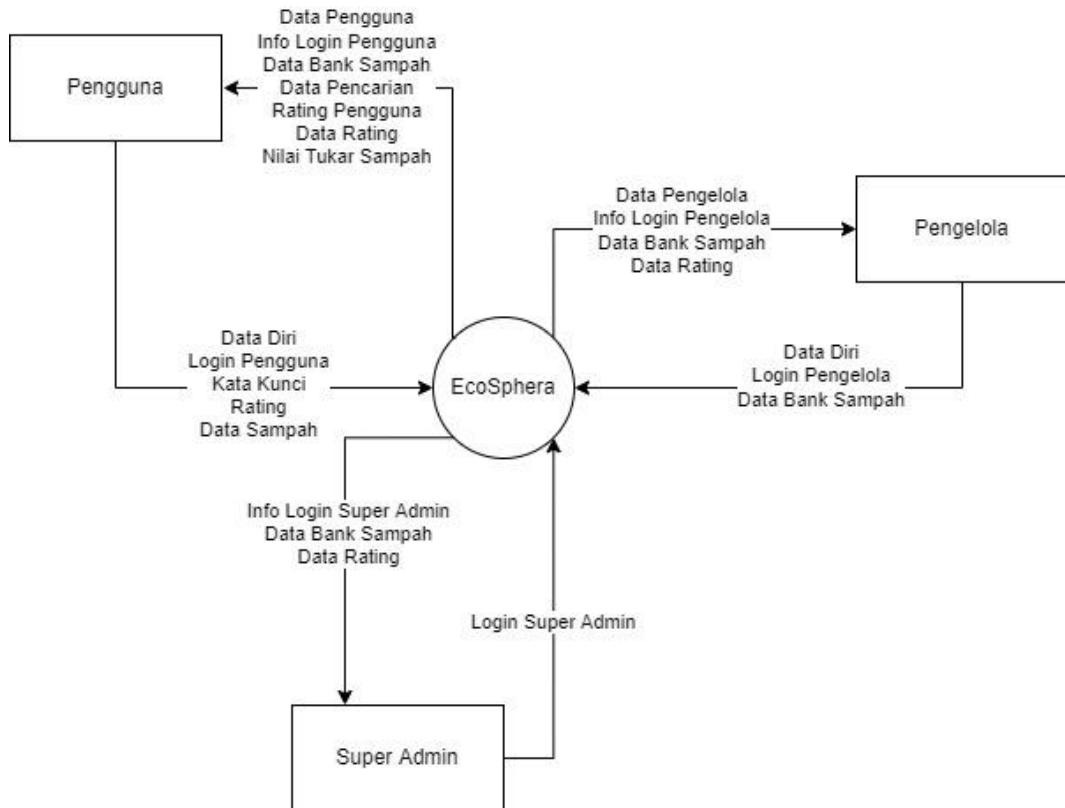
- a. Pengguna diberikan fitur untuk memberikan rating kepada pengelola bank sampah.
- b. Pengguna memilih fitur tersebut kemudian memberikan rating
- c. Setelah memberikan rating, pengguna menyimpan data rating.
- d. Kemudian ditampilkan data hasil rating pengguna tersebut untuk pengelola.



Gambar 4.7 Flowmap Rating Bank Sampah

## 5. Data Flow Diagram

### 5.1 Diagram Context



Gambar 5.1 Diagram Context

#### a. Pengguna

- 1) Memasukkan data diri dan menghasilkan output data pengguna untuk pengguna.
- 2) Pengguna melakukan login dan menghasilkan output info login pengguna untuk pengguna.
- 3) Selanjutnya pengguna memasukkan kata kunci dan menghasilkan data pencarian untuk pengguna.
- 4) Kemudian pengguna memberikan data sampah yang menghasilkan output nilai tukar sampah.
- 5) Pengguna mendapatkan output data bank sampah dari pengelola.
- 6) Pengguna dapat memberikan rating dan menghasilkan output rating pengguna untuk pengguna dan menghasilkan output data rating untuk pengguna, pengelola, dan super admin.

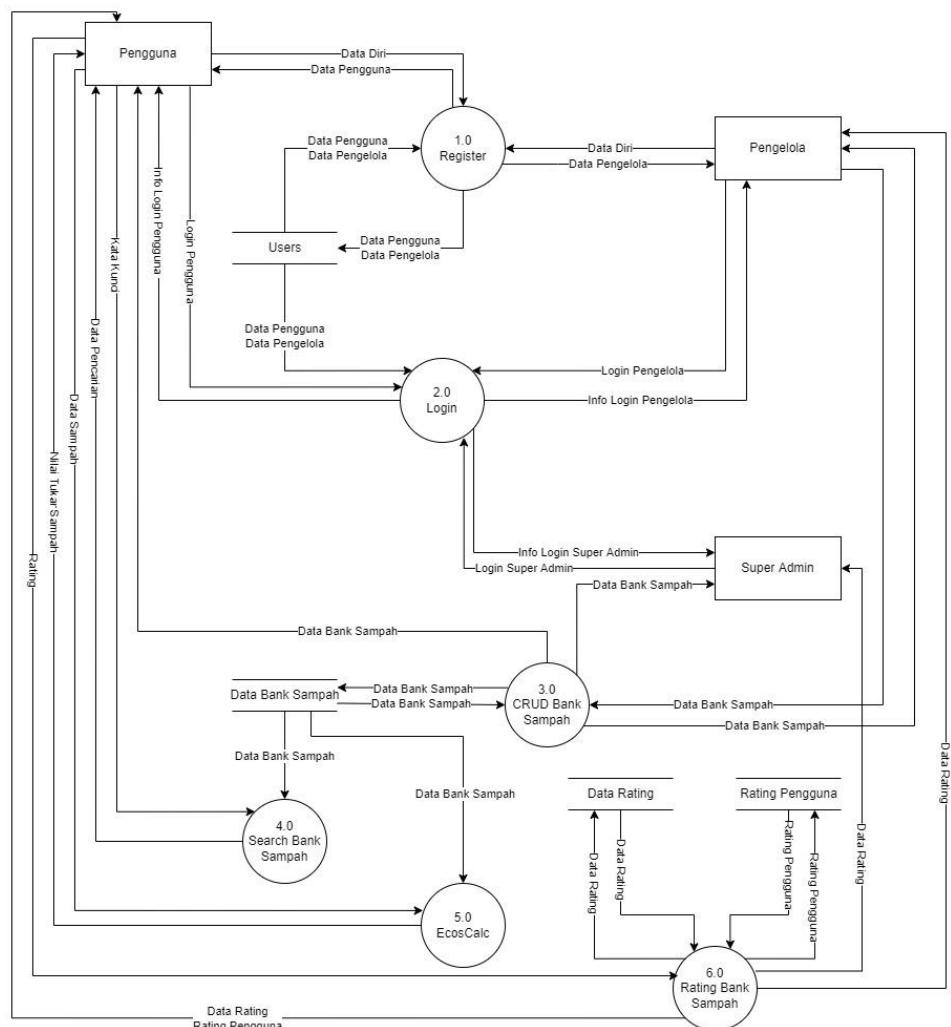
b. Pengelola

- 1) Memasukkan data diri dan menghasilkan output data pengelola.
- 2) Melakukan login pengelola dan menghasilkan output info login pengelola.
- 3) Pengelola memasukkan data bank sampah dan menghasilkan output data bank sampah kepada pengelola, pengguna, dan super admin.
- 4) Pengelola mendapatkan output data rating dari pengguna.

c. Super admin

- 1) Super admin melakukan login super admin dan menghasilkan output info login super admin.
- 2) Super admin akan menerima output data dari pengguna dan pengelola, yaitu data bank sampah dan data rating.

## 5.2 Diagram DFD Level 1



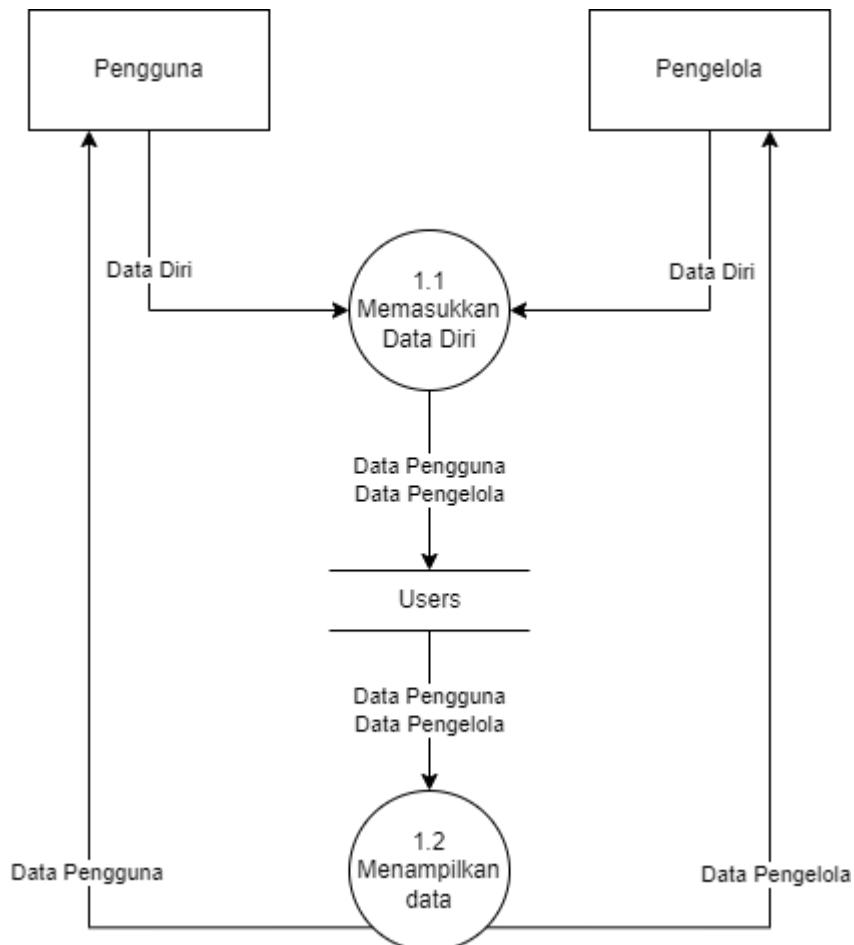
Gambar 5.2 Diagram DFD Level 1

- a. Pengguna
  - 1) Pengguna memasukkan data diri untuk registrasi menghasilkan output data pengguna untuk pengguna.
  - 2) Data pengguna disimpan dalam data store untuk dilanjutkan proses login untuk pengguna.
  - 3) Pengguna melakukan login pengguna dan menghasilkan info login pengguna.
  - 4) Pengguna mendapatkan data bank sampah dari pengelola pada proses CRUD bank sampah
  - 5) Pada proses CRUD Bank Sampah pengguna mendapatkan output data bank sampah dari pengelola.
  - 6) Selanjutnya proses search bank sampah, pengguna memasukkan kata kunci dan menghasilkan output data pencarian.
  - 7) Proses selanjutnya 6.0, yaitu pengguna memasukkan data sampah untuk proses EcosCalc dan menghasilkan output nilai tukar sampah.
  - 8) Pengguna memberikan rating bank sampah dan diproses menjadi rating pengguna, rating pengguna disimpan dalam sistem kemudian menghasilkan output rating pengguna.
  - 9) Pengguna mendapatkan output data rating dari proses data rating bank sampah.
- b. Pengelola
  - 1) Pengelola memasukkan data diri untuk registrasi menghasilkan output data pengelola untuk pengelola.
  - 2) Data pengelola disimpan dalam data store untuk dilanjutkan proses login untuk pengelola.
  - 3) Pengelola melakukan login pengelola dan menghasilkan info login pengelola.
  - 4) Selanjutnya pengelola memasukkan data bank sampah yang di proses pada CRUD bank sampah dan menghasilkan output data bank sampah untuk pengelola.
  - 5) Data bank sampah disimpan pada data store menjadi data pencarian untuk di input pada proses search bank sampah untuk pengguna.

- 6) Data bank sampah juga dimasukkan pada proses ecoscalc yang menghasilkan output nilai tukar sampah untuk pengguna.
  - 7) Setelah itu pengelola mendapatkan output data rating dari pengguna pada proses rating bank sampah.
- c. Super Admin
- 1) Super admin melakukan login super admin pada proses login dan menghasilkan output info login super admin.
  - 2) Super admin mendapatkan output data bank sampah pada proses CRUD bank sampah dari pengelola.
  - 3) Selanjutnya super admin juga mendapatkan output data rating dari pengguna pada proses rating bank sampah.

### 5.3 Diagram DFD Level 2

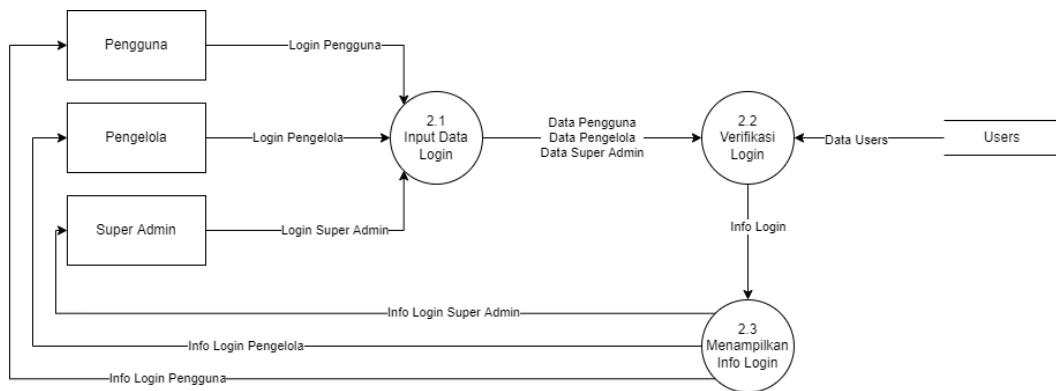
#### 5.3.1 Diagram DFD Level 2 Register



Gambar 5.3.1 Diagram DFD Level 2 Register

- 1) Pengguna dan pengelola memasukkan data diri menjadi output data pengguna dan data diri.
- 2) Data disimpan pada data store kemudian data di proses untuk ditampilkan pada proses 1.2 dan menghasilkan output data pengguna untuk pengguna, data pengelola untuk pengelola.

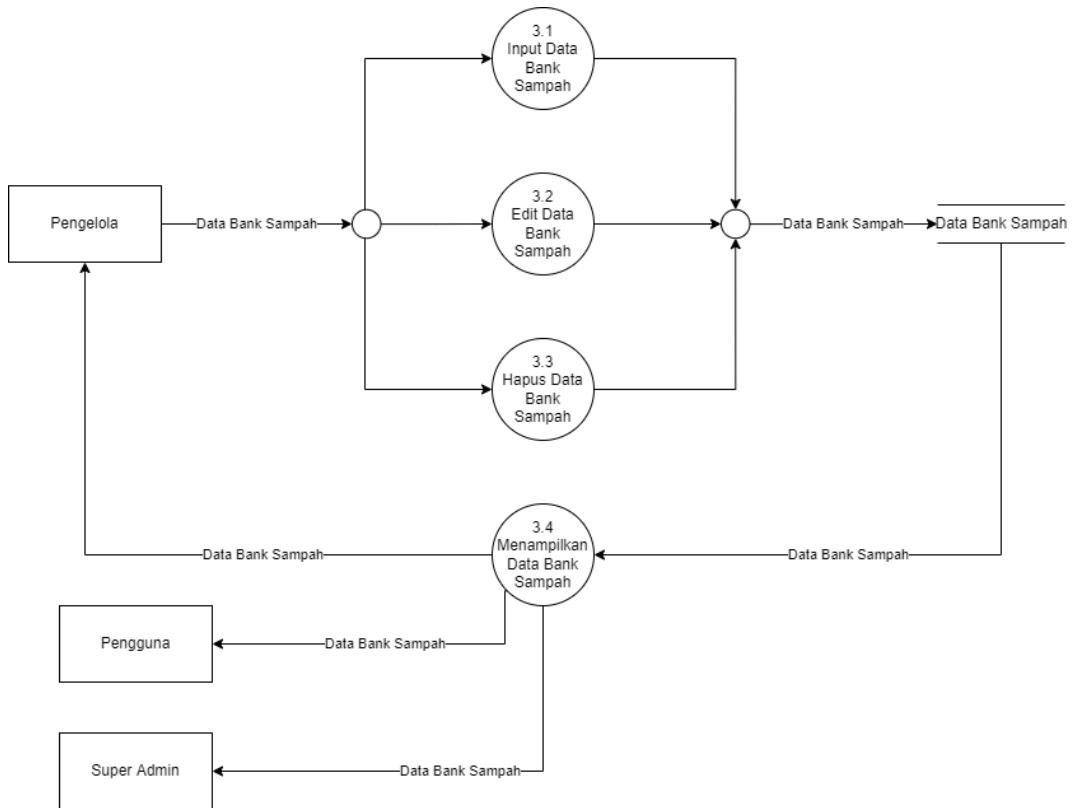
### 5.3.2 Diagram DFD Level 2 Login



Gambar 5.3.2 Diagram DFD Level 2 Login

- 1) Untuk pengguna melakukan login pengguna, pengelola melakukan login pengelola, dan super admin melakukan login super admin.
- 2) Dilakukan proses input data login pada proses 2.1 dan menghasilkan output data pengguna, data pengelola, dan super admin.
- 3) Dilakukan proses 2.2 verifikasi login dengan input data pengguna, data pengelola, dan data super admin.
- 4) Selain itu data user yang disimpan pada data store di input pada proses verifikasi login tersebut.
- 5) Proses verifikasi login menghasilkan output info login untuk diinputkan kembali pada proses 2.3 menampilkan info login.
- 6) Proses menampilkan info login menghasilkan output info login pengguna untuk pengguna, info login pengelola untuk pengelola, dan info login super admin untuk super admin.

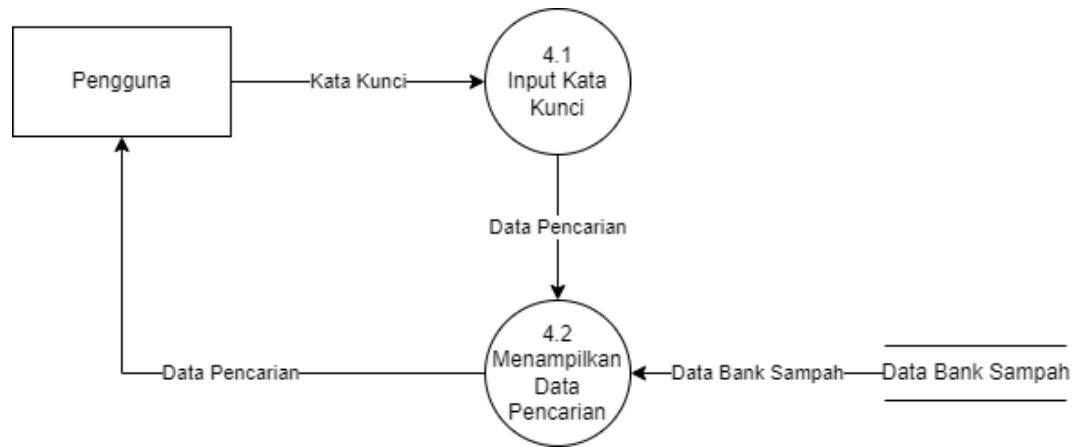
### 5.3.3 Diagram DFD Level 2 CRUD Bank Sampah



Gambar 3.3 Diagram DFD Level 2 CRUD Bank Sampah

- 1) Pengguna memasukkan data bank sampah untuk proses 3.1 input data bank sampah, 3.2 edit data bank sampah, dan 3.3 hapus data bank sampah.
- 2) Proses tersebut menghasilkan output data bank sampah, kemudian disimpan menjadi data bank sampah.
- 3) Kemudian data bank sampah dimasukkan pada proses 3.4 menampilkan data bank sampah dan menghasilkan output data bank sampah untuk pengguna, pengelola, dan super admin.

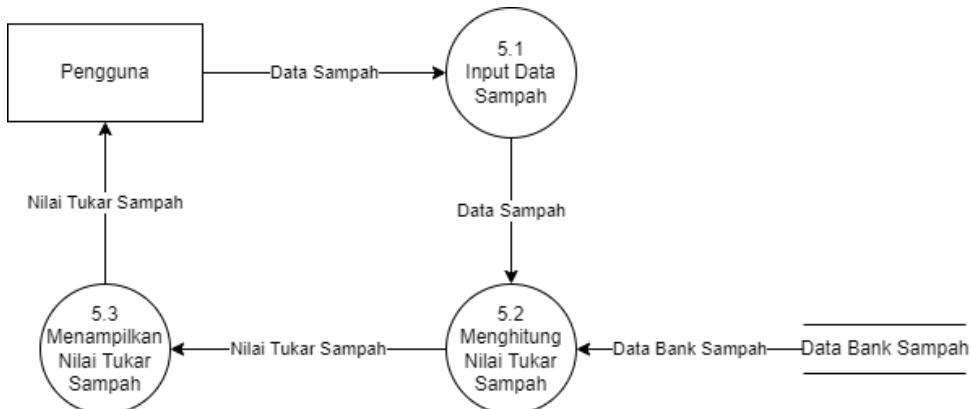
### 5.3.4 Diagram DFD Level 2 Search Bank Sampah



Gambar 3.4 Diagram DFD Level 2 Search Bank Sampah

- 1) Pengguna memasukkan kata kunci untuk diproses pada 4.1 Input kata kunci dan menghasilkan output data pencarian.
- 2) Data pencarian dimasukkan ke dalam proses 4.2 menampilkan data pencarian.
- 3) Data bank sampah yang disimpan pada data bank sampah pun dimasukkan pada proses 4.2 menampilkan data pencarian.
- 4) Proses menampilkan data pencarian menghasilkan output data pencarian untuk pengguna.

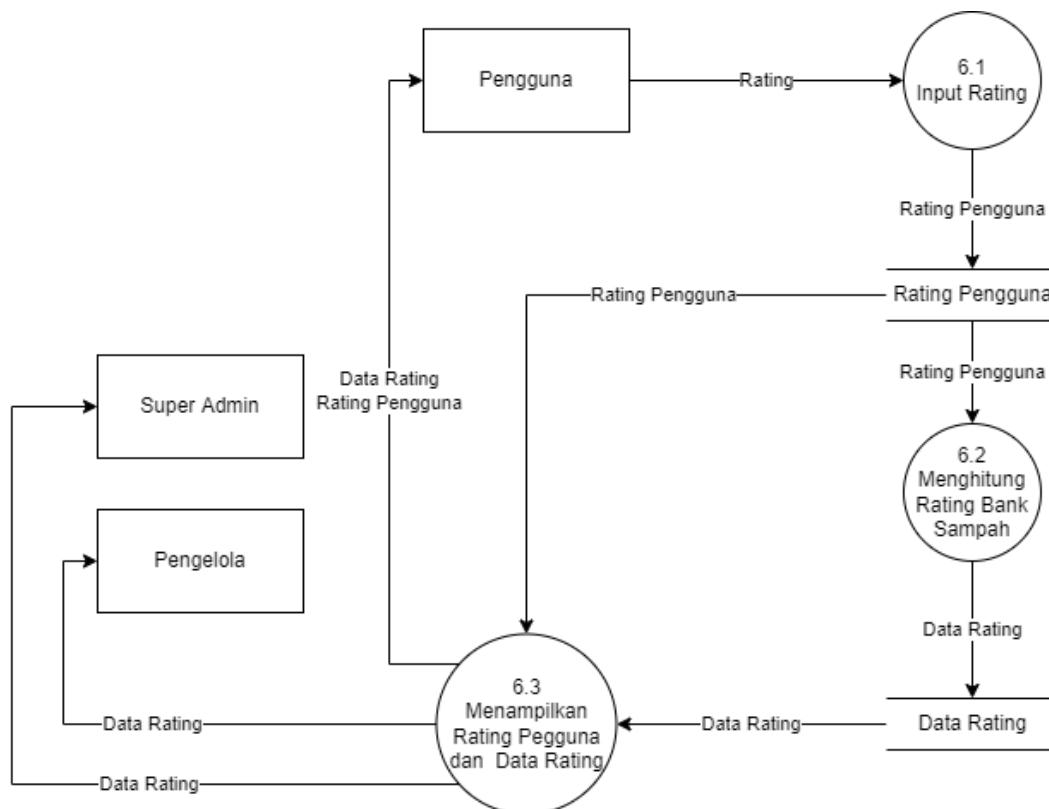
### 5.3.5 Diagram DFD Level 2 EcosCalc



Gambar 5.3.5 Diagram DFD Level 2 EcosCalc

- 1) Pengguna memasukkan data sampah pada proses 5.1 input data sampah dan menghasilkan output data sampah.
- 2) Data sampah tersebut diinput kembali pada proses 5.2 menghitung nilai tukar sampah dan data bank sampah yang disimpan di input juga pada proses menghitung nilai tukar sampah.
- 3) Proses menghitung nilai tukar sampah menghasilkan output nilai tukar sampah.
- 4) Kemudian nilai tukar sampah diinput untuk proses 5.3 menampilkan nilai tukar sampah dan menghasilkan output nilai tukar sampah untuk pengguna.

### 5.3.6 Diagram DFD Level 2 Rating Bank sampah



Gambar 5.3.6 Diagram DFD Level 2 Rating Bank Sampah

- 1) Pengguna memberikan rating untuk diproses pada 6.1 input rating dan menghasilkan output rating pengguna.
- 2) Data disimpan pada data store kemudian rating pengguna diinput untuk proses 6.2 menghitung rating bank sampah dan menghasilkan output data rating.
- 3) Data rating disimpan dan diinput untuk proses 6.3 menampilkan rating pengguna dan data rating menghasilkan output data rating pengguna untuk pengguna serta data rating untuk pengelola dan super admin.

#### **5.4 Kamus Data**

Nama	:	Register
Digunakan Pada	:	Proses Register (Pendaftaran)
Deskripsi	:	Register = Username + Password Username: *String* Password: *String*
Nama	:	Login
Digunakan Pada	:	Proses Login (Masukkan)
Deskripsi	:	Login = Username + Password Username: *String* Password: *String*
Nama	:	CRUD Bank Sampah
Digunakan Pada	:	Proses Create, Read, Update, dan Delete
Deskripsi	:	CRUD = Nama Bank Sampah + Nama Pengelola + Lokasi Bank Sampah + Link Gmaps + Jenis dan Nilai Tukar Sampah Nama Bank Sampah: *String* Nama Pengelola: *String* Lokasi Bank Sampah: *String* Link Gmaps: *Text*
Nama	:	Search

Digunakan Pada	:	Proses Pencarian Bank Sampah
Deskripsi	:	Search = Kata Kunci + Data Bank Sampah + Data Pencarian Kata Kunci: *String* Data Bank Sampah: *String* Data Pencarian: *String*
Nama	:	EcosCalc
Digunakan Pada	:	Proses Kalkulasi Nilai Tukar sampah
Deskripsi	:	Ecoscalc = Jenis Sampah + Berat Sampah + Data Nilai Tukar Jenis Sampah: *String* Berat Sampah: *Integer* Data Nilai Tukar: *Integer*
Nama	:	Rating Bank Sampah
Digunakan Pada	:	Proses Penilaian Terhadap Kualitas Bank Sampah
Deskripsi	:	Rating Bank Sampah = Rating + Rating Pengguna + Data Rating Rating: *Integer* Rating Pengguna: *Integer* Data Rating: *Integer*

## 6. Project Methodology

Dalam perancangan EcoSpehra, sistem yang akan digunakan yaitu metode Agile dengan framework Scrum. Karena dengan metode ini EcoSphera dapat memberikan perubahan apabila terdapat perubahan mendadak dari stakeholders. Berikut beberapa alasan mengapa EcoSphera memilih metode Agile:

- a. Fleksibel terhadap perubahan: Dalam proyek ini, EcoSphera memiliki kemungkinan untuk berubah sewaktu-waktu sesuai dengan teknologi yang baru. Oleh karena itu, metode agile sangat cocok dengan proyek EcoSphera agar tim pengembang dapat menyesuaikan kebutuhan pengguna.

- b. Kolaborasi terus-menerus: Dalam proyek ini, tim pengembang EcoSphera akan berkolaborasi secara terus-menerus dengan cara rapat harian antar tim, mengundang stakeholders untuk melihat hasil kerja tim pengembang sehingga tim pengembang dapat langsung menerima umpan balik. Dengan begitu, metode agile ini dapat membuat tim pengembang mendapatkan kemudahan dalam menentukan kebutuhan pengguna.
- c. Peningkatan kepuasan pengguna: Dengan metode ini, kepuasan pengguna akan meningkat karena pengguna dapat melihat perkembangan EcoSphera sesuai dengan yang mereka inginkan.
- d. Proyek yang relevan: Dengan metode ini EcoSphera akan sangat relevan dengan pengguna. Karena, tim pengembang mengembangkan EcoSphera berdasarkan prioritas yang dibutuhkan pengguna.

## 7. Project Schedule

No	Jadwal	Aktivitas	Penanggungjawab
1	09 September – 30 September 2024	Analisis Kebutuhan	Rafi Islami Pasha Dini Hari Putra
2	30 September – 07 Oktober 2024	Pembuatan Flowmap	Rafi Islami Pasha Dini Hari Putra
3	07 Oktober 2024 – 19 Oktober 2024	Pembuatan Data Flow Diagram,	Rafi Islami Pasha Dini Hari Putra
4	11 November 2024 – 20 Oktober 2024	Pembuatan Kamus Data, Project Methodology, Project Schedule, dan Project Budget	Rafi Islami Pasha Dini Hari Putra
5	21 Oktober - 03 November 2024	Pembuatan Design Architecture, Design System, Wireframe, User Interface	Rafi Islami Pasha Dini Hari Putra
6	21 November – 03 Desember 2024	Implementasi Kodingan	Rafi Islami Pasha Dini Hari Putra

7	26 November – 03 Desember 2024	Update SKPL	Rafi Islami Pasha Dini Hari Putra
8	22 Desember – 03 Desember	Pengujian	Rafi Islami Pasha Dini Hari Putra

### 7.1 Analisis Kebutuhan

- a. Menentukan ide
- b. Mengidentifikasi kendala
- c. Menentukan ruang lingkup proyek
- d. Menentukan jenis kebutuhan fungsional dan nonfungsional
- e. Melakukan elicitation dengan wawancara bersama stakeholder
- f. Dokumentasi kebutuhan
- g. Negosiasi kebutuhan dengan stakeholder
- h. Analisis dan pembuatan SKPL

### 7.2 Pembuatan Flowmap

- Melakukan pembuatan Flowmap untuk menggambarkan setiap Langkah secara rinci yang terbagi ke dalam beberapa unit yang menjalankan proses tersebut.
- Memberikan hasil flowmap sementara ke dalam dosen pembimbing untuk dilakukan revisi
- Melakukan revisi pembuatan flowmap
- Menambahkan flowmap ke dalam dokumen srs dan memberikan penjelasan mengenai setiap alur.

### 7.3 Pembuatan DFD (Data Flow Diagram)

- a. Membuat DFD untuk menggambarkan aliran data antara proses dan penyimpanan data tersebut.
- b. Menambahkan DFD ke dalam dokumen SRS dan memberikan penjelasan mengenai setiap alur.

## **7.4 Update SKPL**

- a. Merevisi Flowmap dan Data Flow Diagram (DFD) pada bagian-bagian yang belum tepat sehingga sesuai dengan kebutuhan berdasarkan hasil diskusi dengan stakeholder. Kemudian meninjau ulang untuk memastikan konsistensi dan kejelasan.
- b. Menambahkan kamus data, project methodology, dan project budget pada dokumen SKPL

## **7.5 Pembuatan Design EcoSphera**

- a. Membuat *Design System*, diantaranya *color pallete*, *typography*, *icon*, *component button*, dan *input*. *Design system* bermanfaat untuk memudahkan dan mempercepat proses kerja.
- b. Pembuatan *Wireframe* untuk memberikan kerangka awal dalam menentukan tata letak (layout) agar memperjelas arah dan tujuan. Pada proses ini belum difokuskan pada gambaran detail, seperti *icon* ataupun *component* lainnya.

## **7.6 Implementasi Kodingan**

- a. Menentukan bahasa pemrograman yang akan dipakai dalam pembuatan project ini. Pada project ini, menggunakan bahasa pemrograman python dan library pandas untuk pengelolaan data yang diperlukan
- b. Pembuatan kode untuk fitur-fitur yang akan diimplementasikan, diantaranya login, register, search bank sampah, CRUD, dan fitur rating.
- c. Pembuatan dataset menggunakan pandas untuk menyimpan data bank sampah, users, dan rating.

## 8. Project Budget

Item	Deskripsi	Estimasi Biaya
Personel	Gaji, tunjangan pengembang 2 UI/UX Frontend 3 Backend	$2 \times 5.000.000 = 10.000.000$ $3 \times 6.000.000 = 18.000.000$
Alat Pengembangan dan Dokumentasi	Software Pro dan Dokumentasi Pengembangan	1.000.000
Pengujian	Biaya pengujian kualitas dan bug fixing	1.000.000
Hosting	Rincian biaya Domain: 200.000/tahun Server: 700.000/tahun	$200.000 + 700.000 = 900.000$
Pemeliharaan	Dukungan teknis dan update sistem	1.500.000
Lain-lain	Keperluan tambahan (backup, dokumen)	1.000.000
<b>Total Biaya</b>		Rp33.400.000,00

## 9. Software Design

### 9.1 Architecture Design

Model View Controller (MVC) adalah sebuah pola arsitektur dalam membuat sebuah aplikasi dengan cara memisahkan kode menjadi tiga bagian yang terdiri dari:

- **Model**

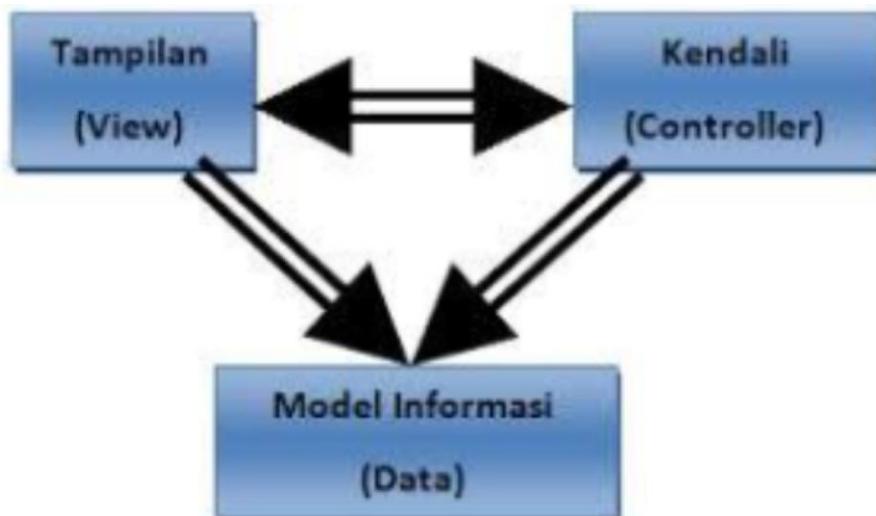
Bagian yang bertugas untuk menyiapkan, mengatur, memanipulasi, dan mengorganisasikan data yang ada di database.

- **View**

Bagian yang bertugas untuk menampilkan informasi dalam bentuk *Graphical User Interface* (GUI).

- **Controller**

Bagian yang bertugas untuk menghubungkan serta mengatur model dan view agar dapat saling terhubung.



Sumber : J Deacon (2009)

Gambar 9.1 Architecture Design MVC

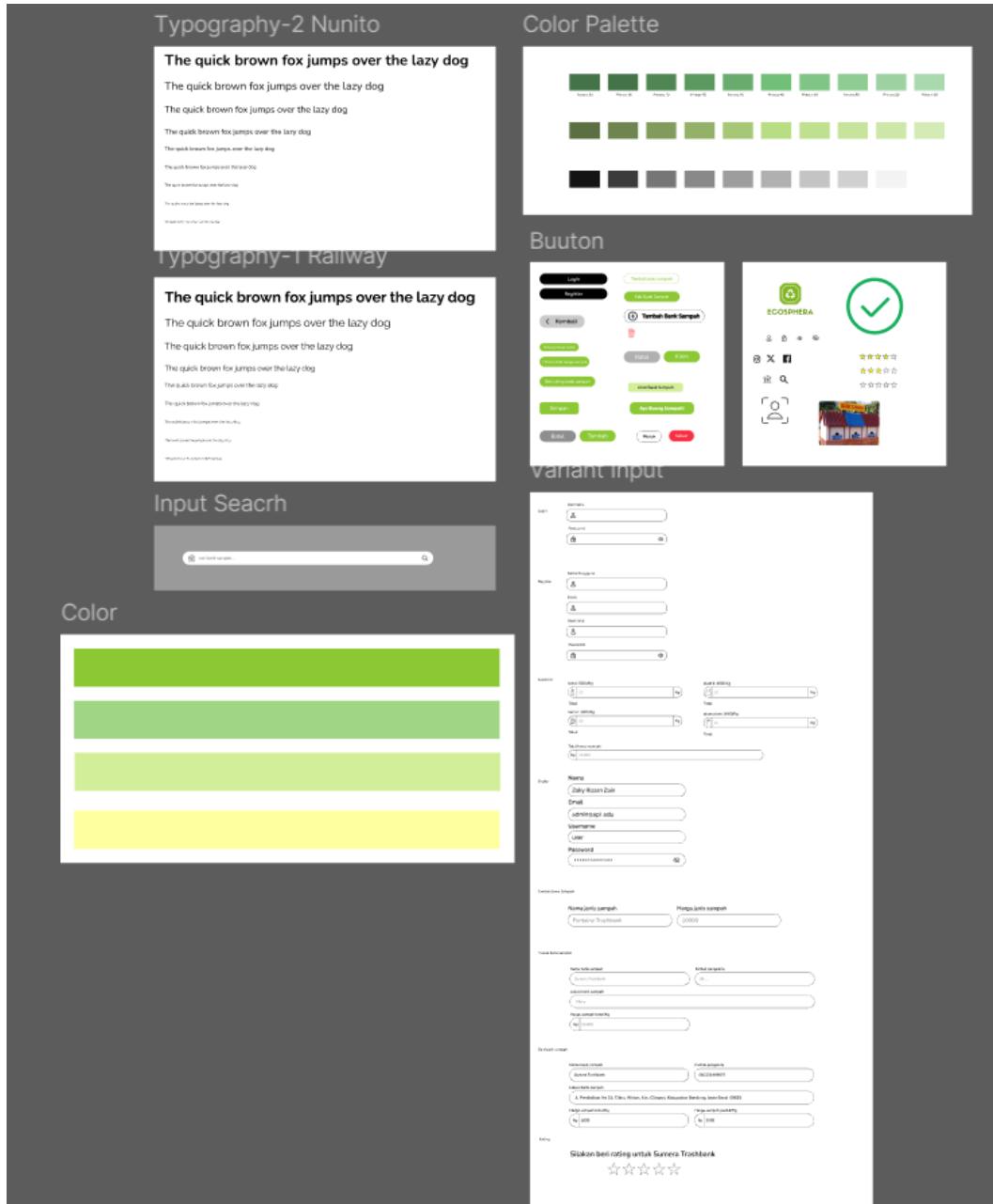
Alur design architecture MVC, user dapat memilih untuk request kepada view atau controller.

- a. User melakukan request kepada view.
  - 1) Proses dimulai dengan user request kepada view.
  - 2) Kemudian view request kepada controller untuk mengirimkan data yang diminta oleh user.
  - 3) Lalu controller melakukan request kepada model untuk meminta data.
  - 4) Dilanjutkan kembali oleh model untuk meminta apakah data tersebut ada pada database.
  - 5) Database memberikan respon dan memberikan datanya kepada model.
  - 6) Selanjutnya model meneruskan respon kepada controller.
  - 7) Dilanjutkan dengan controller memberikan respon kepada view.

- 8) View memberikan respon kepada user untuk menampilkan tampilan pada view.
- b. User melakukan request kepada controller.
  - 1) Dimulai dengan user request kepada controller.
  - 2) Selanjutnya controller request kepada model untuk mengirimkan data yang diminta oleh user.
  - 3) Dilanjutkan kembali oleh model untuk meminta apakah data tersebut ada pada database.
  - 4) Database memberikan respon dan memberikan datanya kepada model.
  - 5) Kemudian model meneruskan respon kepada controller.
  - 6) Controller memberikan respon kepada user untuk menampilkan tampilan pada controller.

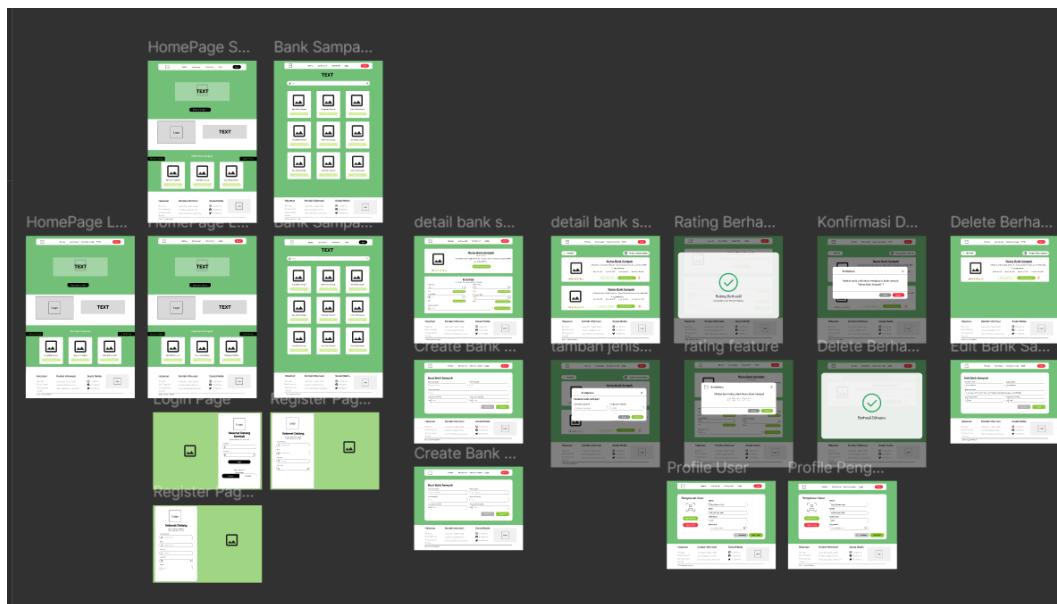
## 9.2 User Interface Design

### 9.2.1 Design System



Gambar 9.2.1 Design System

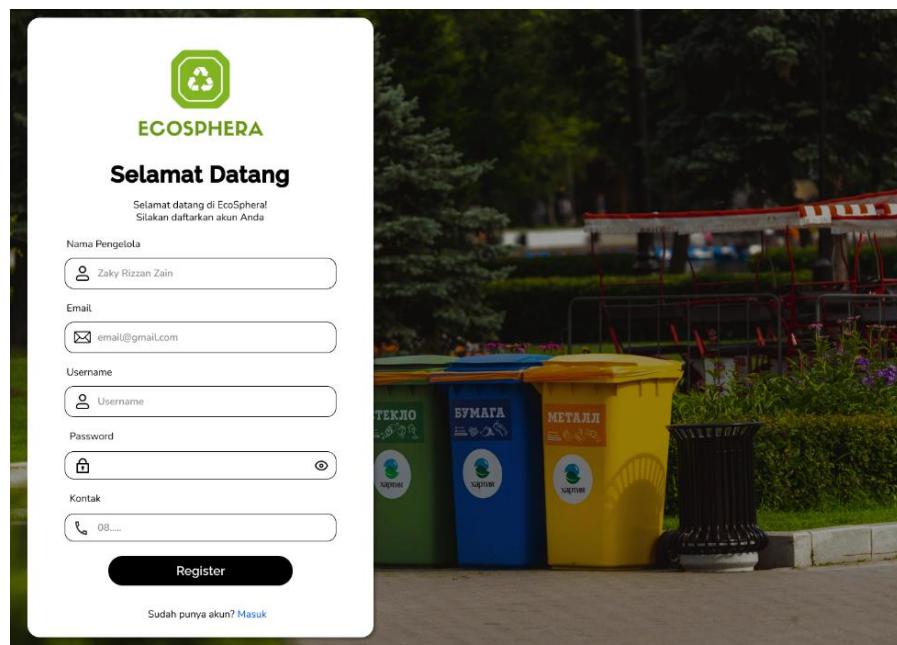
## 9.2.2 Wireframe



Gambar 9.2.2 Wireframe

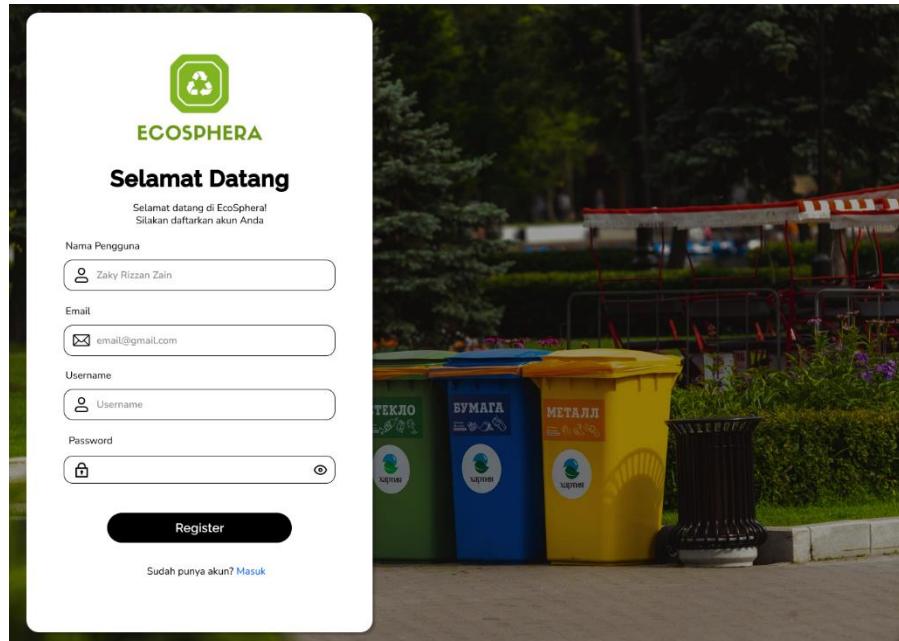
## 9.2.3 User Interface

### a. Halaman Registrasi Pengelola



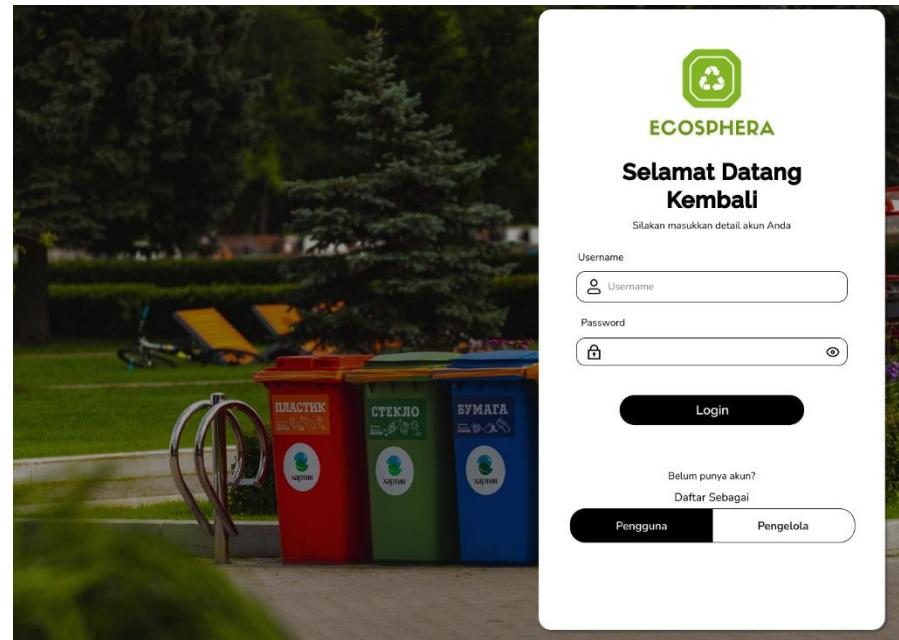
Gambar a. Halaman Register Pengelola

b. Halaman Register Pengguna



Gambar b. Halaman Register Pengguna

c. Halaman Login:



Gambar c. Halaman Login

#### d. Halaman Dashboard Pengguna

**Solusi Digital Bank Sampah**

Ecosphaera menghadirkan solusi digital yang memudahkan pengelolaan sampah anorganik secara efisien dan terintegrasi.

**Jaga Lingkungan sekitarmu tetap bersih!**

Ecosphaera berusaha untuk meminimalkan penggunaan sampah langsung dari sumbernya. Dengan memanfaatkan platform digital, Ecosphaera menghubungkan pemiliknya dengan jaringan pengelolaan sampah lokal dan internasional. Selain itu, Ecosphaera juga memberikan dukungan teknologi terkait dalam menjalankan bisnis sampah.

**Jenis Sampah**

Ecosphaera mewajibkan para pengelola sampah untuk bertujuan jenis sampah yang dapat dicuci ulang

- Kertas
- Plastik
- Botol
- aluminum

**Bank Sampah dengan Rating Terbaik**

**Ayo Kelola Sampah dengan Mudah Bersama Sumera Bank!**

Sumera Bank hadir untuk mempermudah colok pada saat pengelolaan sampah. Dengan layanan digital yang terintegrasi, kamu memudahkan diri untuk memilah, mengelola, dan mendaur ulang sampah anorganik secara efisien. Melalui kemitraan dengan bank sampah lokal, Anda dapat mengirimkan sampah ke titik pengumpulan terdekat dengan mudah.

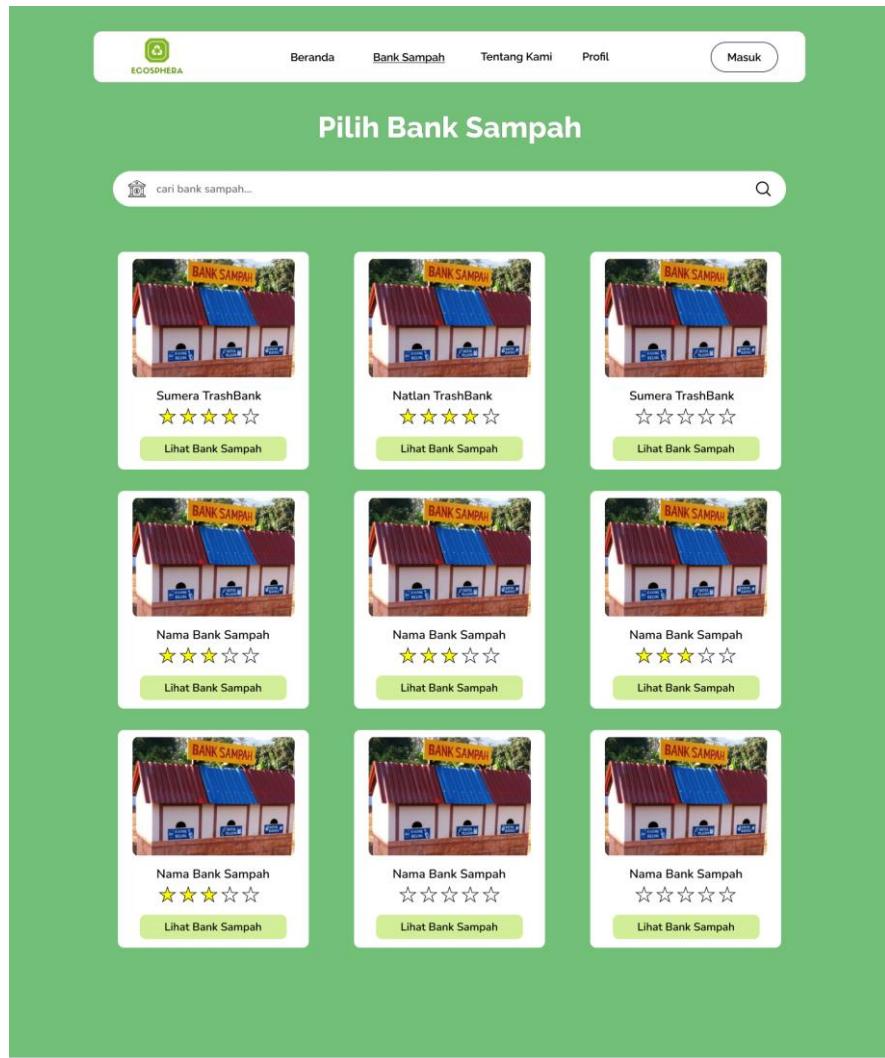
Lihat Bank Sampah

Halaman	Kontak Informasi	Sosial Media
Beranda Bank Sampah Tentang Kami Kontak	(+62) 821-2394-7660 ecosphere@gmail.com Cibiru Bandung, West Java	@ecosphere... EcoSphere EcoSphera

©2024 Copyright Ecosphaera

Gambar d. Halaman Dashboard Pengguna

### e. Halaman Bank Sampah



#### Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

#### Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

#### Sosial Media

ecosphera\_  
 EcoSphera  
 EcoSphera\_

©2024 Copyright EcoSphera

Gambar e. Halaman Bank Sampah

## f. Halaman Bank Sampah Pengelola

The screenshot shows a mobile-optimized web page for selecting waste banks. At the top, there's a header with the Ecosphaera logo, navigation links for 'Beranda', 'Bank Sampah', 'Kelola bank sampah', 'Profile', and a 'Masuk' button. Below the header is a search bar with a placeholder 'cari bank sampah...' and a magnifying glass icon. The main content area is titled 'Pilih Bank Sampah' and displays a grid of nine waste bank entries, each featuring a small image of a waste bank building, the bank's name, a star rating, and a 'Lihat Bank Sampah' button.

Bank Sampah	Rating	Action
Sumera TrashBank	★★★★★	Lihat Bank Sampah
Natlan TrashBank	★★★★★	Lihat Bank Sampah
Sumera TrashBank	★★★★★	Lihat Bank Sampah
Nama Bank Sampah	★★★★★	Lihat Bank Sampah
Nama Bank Sampah	★★★★★	Lihat Bank Sampah
Nama Bank Sampah	★★★★★	Lihat Bank Sampah
Nama Bank Sampah	★★★★★	Lihat Bank Sampah
Nama Bank Sampah	★★★★★	Lihat Bank Sampah

### Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

### Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

### Sosial Media

ecosphaera\_  
 EcoSphera  
 EcoSphera\_



©2024 Copyright EcoSphera

Gambar. Halaman Bank Sampah Pengelola

### g. Halaman Detail Bank Sampah

The screenshot shows the EcoSphera website's 'Bank Sampah' page. At the top, there is a navigation bar with links to Beranda, Bank Sampah (which is underlined to indicate it's the current page), Tentang Kami, Profil, and Keluar. Below the navigation is a header for 'Sumera Trashbank' with the address 'Nahida-Chan Jl. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625' and phone number '082233445678'. There is also a button to 'Beri rating bank sampah'. To the left, there is a photo of a building with a red roof and a sign that says 'BANK SAMPAH'. Below the photo is a rating of '4.0 ★★★★☆'. The main content area is titled 'EcosCalc' and contains four input fields for calculating the value of different types of waste:

- botol: 5000/Kg (input field: 10 Kg, button: Hitung harga botol)
- plastik: 4000/Kg (input field: 10 Kg, button: Hitung harga plastik)
- kertas: 3000/Kg (input field: 10 Kg, button: Hitung harga kertas)
- alumunium: 9000/Kg (input field: 10 Kg, button: Hitung harga alumunium)

At the bottom, there is a summary line: 'Total harga sampah Rp 10.000' with a button to 'Hitung total harga sampah'.

#### Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

#### Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, Jawa Barat

#### Sosial Media

ecosphera\_  
 EcoSphera  
 EcoSphera\_



©2024 Copyright EcoSphera

*Gambar. Halaman Detail Bank Sampah*

## h. Halaman Detail Bank Sampah Pengelola

The screenshot shows a green-themed web interface for managing waste banks. At the top, there's a navigation bar with the EcoSphera logo, menu items like Beranda, Bank Sampah, Kelola bank sampah, Profile, and a Keluar button. Below the navigation is a back button labeled 'Kembali'. The main content area displays two entries for waste banks:

- Sumera Trashbank**: Located at Jl. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625. Contact number: 082233445678. Services: Botol: Rp. 3.000, Kertas: Rp. 2.000, Kertas: Rp. 2.000, Aluminium: Rp. 9.000. Rating: 4.0 stars.
- Fontaine Bank**: Located at Jl. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625. Contact number: 085698674523. Services: Botol: Rp. 3.000, Kertas: Rp. 2.000, Kertas: Rp. 3.000, Aluminium: Rp. 9.000. Rating: 3.0 stars.

### Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

### Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

### Sosial Media

ecosphera\_  
 EcoSphera  
 EcoSphera\_



©2024 Copyright EcoSphera

Gambar h. Halaman Detail Bank Sampah Pengelola

### i. Halaman Detail Kelola Bank Sampah

The screenshot displays the 'Kelola bank sampah' section of the EcoSphera application. It shows two entries for waste banks:

- Sumera Trashbank**  
Address: JL. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625  
Phone: 082233445678  
Botol: Rp. 3.000    Kertas: Rp. 2.000    Kertas: Rp. 2.000    Aluminium: Rp. 9.000  
Rating: 4.0 ★★★★☆  
Buttons: Tambah jenis sampah, Edit Bank Sampah, Delete
- Fontaine Bank**  
Address: JL. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625  
Phone: 085698674523  
Botol: Rp. 3.000    Kertas: Rp. 2.000    Kertas: Rp. 3.000    Aluminium: Rp. 9.000  
Rating: 3.0 ★★★☆☆  
Buttons: Tambah jenis sampah, Edit Bank Sampah, Delete

#### Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

#### Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

#### Sosial Media

ecosphera\_  
 EcoSphera  
 EcoSphera\_



©2024 Copyright EcoSphera

Gambar i. Halaman Detai Kelolah Bank Sampah

j. Halaman Edit Bank Sampah Pengelola

Edit Bank Sampah

Nama bank sampah  
Sumera Trashbank

Kontak pengelola  
082133445678

Lokasi bank sampah  
JL. Pendidikan No.15, Cibiru Wetan, Kec. Cileunyi, Kabupaten Bandung, Jawa Barat 40625

Harga sampah botol/Kg  
Rp 5000

Harga sampah plastik/Kg  
Rp 3000

Kembali      Ubah

Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

Sosial Media

ecosphera\_  
 EcoSphere  
 EcoSphera\_



©2024 Copyright EcoSphera

Gambar j. Edit Bank Sampah Pengelola

k. Delete Bank Sampah

Kembali      Tambah Bank Sampah

Sumera Trashbank

EcoSphera

Apakah anda yakin akan menghapus bank sampah "Fontaine Bank" ?

Batal      Hapus

Botol: Rp. 3.000	Kertas: Rp. 2.000	Kertas: Rp. 3.000	Aluminium: Rp. 9.000
Tambah jenis sampah      Edit Bank Sampah			

Halaman

Beranda  
Bank Sampah  
Tentang Kami  
Kontak

Kontak Informasi

(+62) 821-2394-7660  
ecosphera@gmail.com  
Cibiru Bandung, West Java

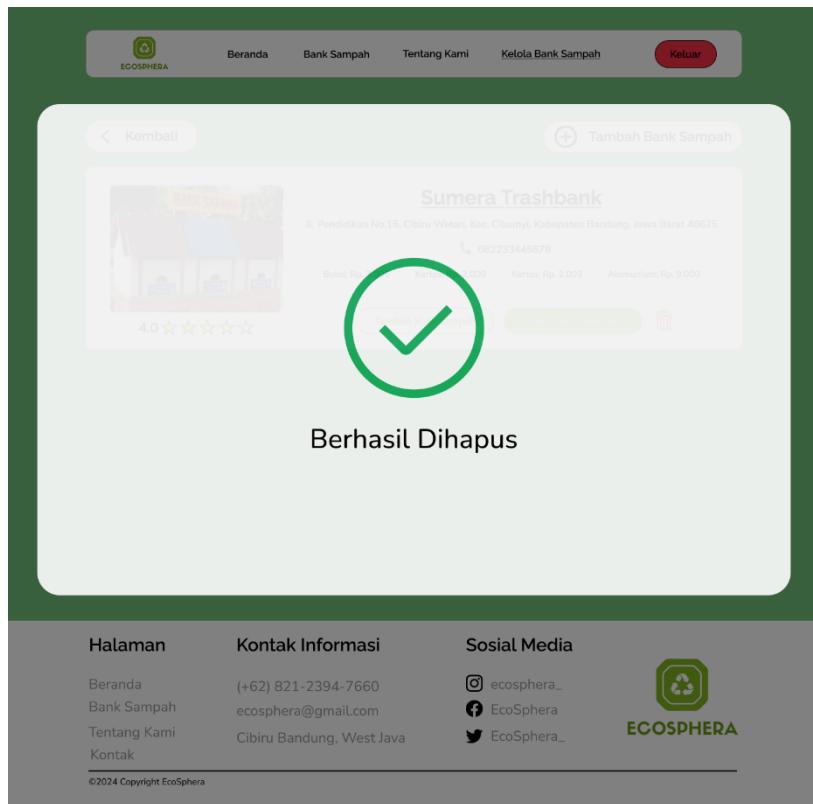
Sosial Media

ecosphera\_  
 EcoSphere  
 EcoSphera\_

©2024 Copyright EcoSphera

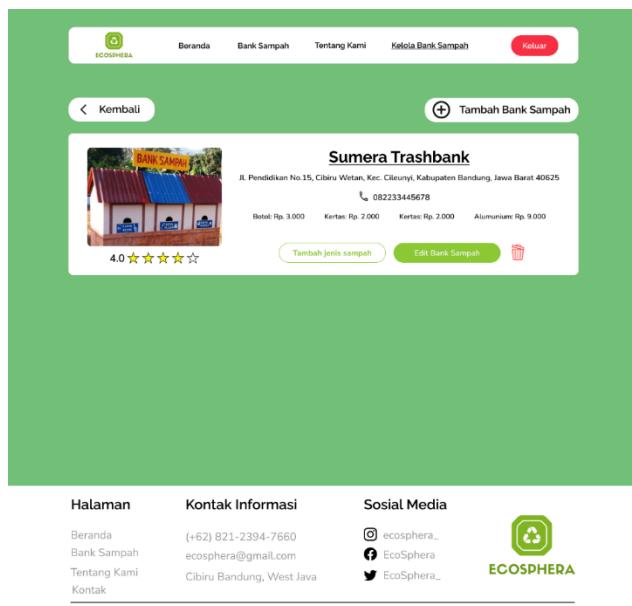
Gambar k. Delete Bank Sampah

## 1. Delete Berhasil 1



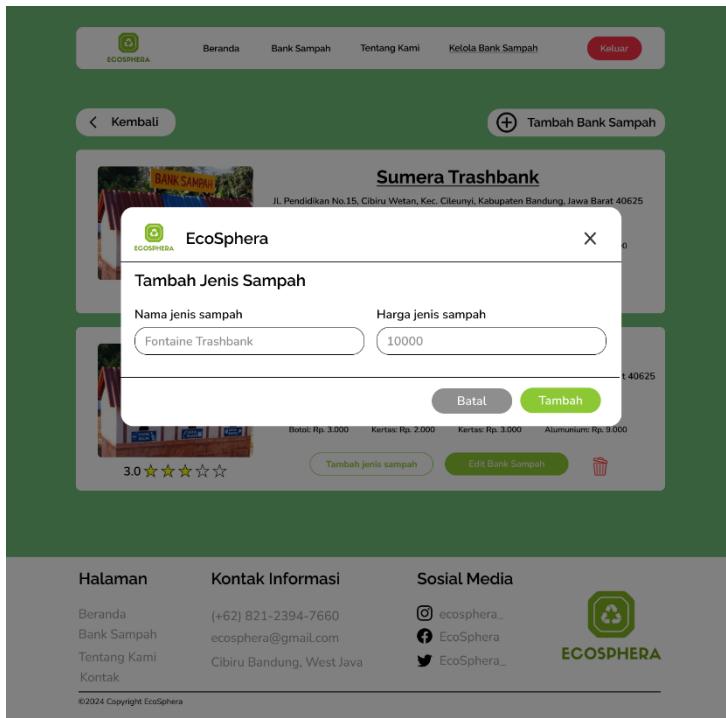
Gambar. Delete Berhasil 1

## m. Delete Berhasil 2



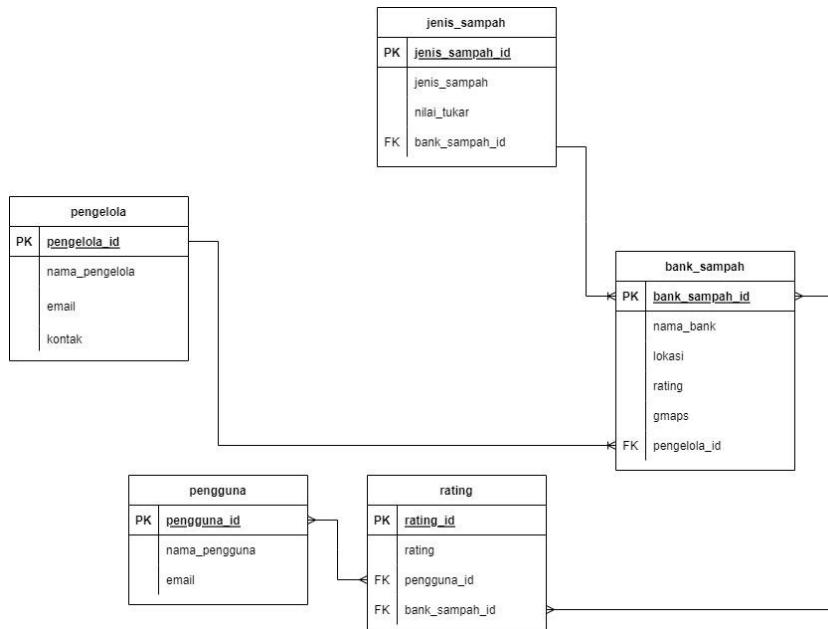
Gambar m. Delete Berhasil 2

## n. Tambah Jenis Sampah



Gambar n. Tambah Jenis Sampah

## 9.2 Database Design



Gambar 9.3 Database Design

a. Tabel pengelola

Relasi dengan bank\_sampah:

One to Many: Setiap pengelola bertanggung jawab atas banyak bank\_sampah  
(melalui Foreign Key pengelola\_id di tabel bank\_sampah).

b. Tabel pengguna

Tidak ada relasi dengan tabel users karena tabel tersebut dihapus. Sebagian gantinya, tabel pengguna berdiri sendiri atau dapat berrelasi langsung dengan tabel lainnya jika diperlukan di masa depan.

c. Tabel bank\_sampah

Relasi dengan pengelola:

Many to One: Banyak bank\_sampah dapat dikelola oleh satu pengelola (melalui Foreign Key pengelola\_id di tabel bank\_sampah).

Relasi dengan jenis\_sampah:

One to Many: Satu bank\_sampah dapat mengelola banyak jenis\_sampah  
(melalui Foreign Key bank\_sampah\_id di tabel jenis\_sampah).

Relasi dengan rating:

One to Many: Satu bank\_sampah dapat menerima banyak rating (melalui Foreign Key bank\_sampah\_id di tabel rating).

d. Tabel jenis\_sampah

Relasi dengan bank\_sampah:

Many to One: Banyak jenis\_sampah dikelola oleh satu bank\_sampah (melalui Foreign Key bank\_sampah\_id).

e. Tabel rating

Relasi dengan pengguna:

Many to One: Banyak rating dapat diberikan oleh satu pengguna (melalui Foreign Key pengguna\_id di tabel rating).

Relasi dengan bank\_sampah:

Many to One: Banyak rating dapat diberikan untuk satu bank\_sampah (melalui Foreign Key bank\_sampah\_id di tabel rating).

## 10. Implementasi

```
import pandas as pd
import re

auth = {'auth': 'user'}
user_login = {'username': ''}

def kelolaBank():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';', dtype={'kontak pengelola': str})
    header = ['nama bank sampah', 'nama pengelola', 'lokasi', 'kontak pengelola', 'harga sampah organik', 'harga sampah non-organik', 'rating']
    bank_sampah_pengelola = bank_sampah[bank_sampah['nama pengelola'] == user_login['username']]

    if(bank_sampah Pengelola.empty):
        step = input('\n[1]Ya\n[2]Tidak\nAnda belum memiliki bank sampah yang dikelola! Apakah ingin membuat bank sampah sekarang? ')
        if(step.replace(' ', '') == ''):
            print('Inputan tidak boleh kosong!')
            kelolaBank()
        elif(not re.match(r'^\d{1}$', step)):
            print('Inputan hanya boleh mengandung angka! Maximal 1 angka!')
            kelolaBank()
        elif(step == '1'):
            nama_bank = input('\nMasukan nama bank sampah: ')
            nama_pengelola = user_login['username']
            lokasi = input('Masukkan lokasi bank sampah: ')
            kontak = input('Masukkan kontak pengelola bank sampah: ')
            organik = input('Masukkan harga sampah organik: ')
            non_organik = input('Masukkan harga sampah non organik: ')
            rating = 0

            if(str(nama_bank).replace(' ', '') == ''):
                print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
                kelolaBank()
            elif(str(lokasi).replace(' ', '') == ''):
                print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
                kelolaBank()
            elif(str(kontak).replace(' ', '') == ''):
                print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
                kelolaBank()
            elif(str(organik).replace(' ', '') == ''):
                print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
                kelolaBank()
            elif(str(non_organik).replace(' ', '') == ''):
                print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
                kelolaBank()
            elif(not re.match(r'^[a-zA-Z\s]{3,}', nama_bank)):
                print('Nama bank sampah hanya boleh berisi huruf dan spasi! Minimal 3 huruf!')
                kelolaBank()
            elif(not re.match(r'^(?:[^a-zA-Z]*[a-zA-Z])\{3\}[a-zA-Z0-9\s,\.\(\)]\{10,\$', lokasi)):
                print('\nLokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurung, setidaknya memiliki 3 huruf! Minimal 10 karakter!')
                kelolaBank()
            elif(not re.match(r'^\d{7,15}\$', kontak)):
                print('\nKontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!')
                kelolaBank()
            elif(not re.match(r'^\d{4}\$', organik)):
                print('\nHarga sampah organik hanya boleh berisi angka, minimal 4 angka!')
                kelolaBank()
            elif(not re.match(r'^\d{4}\$', non_organik)):
                print('\nHarga sampah non organik hanya boleh berisi angka, minimal 4 angka!')
                kelolaBank()
            elif(not re.match(r'^[1-9][0-9]*\$', organik)):
                print('\nHarga sampah organik tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')

            kelolaBank()
            elif(not re.match(r'^[1-9][0-9]*\$', non_organik)):
                print('\nHarga sampah non organik tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
                kelolaBank()

            data = [[nama_bank, nama_pengelola, lokasi, f'{kontak}', organik, non_organik, rating]]

            new_data = pd.DataFrame(data, columns=header)
            new_data.to_csv('data_bank_sampah.csv', mode='a', sep=';', header=False, index=False)

            print('\nData bank sampah berhasil dibuat!')
            kelolaBank()
        elif(step == '2'):
            choicePengelola()
        else:
            print('\nSilakan pilih berdasarkan angka yang tersedia!')
            kelolaBank()
    else:
        print('\n' + 30 * '=')
        choice = input('\n[1]Tambah data bank sampah\n[2]Lihat data bank sampah\n[3]Update data bank sampah\n[4]Hapus data bank sampah\n[5]Kembali\nSilakan pilih fitur berdasarkan angka: ')
```

```

# Validasi choice gaboleh kosong
if(choice.replace(' ', '') == ''):
    print('\nInputan tidak boleh kosong!')
    kelolaBank()

# Validasi choice harus angka dan 1 angka aja
elif(not re.match(r'^\d{1}$', choice)):
    print('\nInput hanya boleh berisi angka! Maximal 1 angka!')
    kelolaBank()

# Create data bank sampah pengelola
elif(choice == '1'):
    nama_bank = input('\nMasukan nama bank sampah: ')
    nama_pengelola = user_login['username']
    lokasi = input('Masukkan lokasi bank sampah: ')
    kontak = input('Masukkan kontak pengelola bank sampah: ')
    organik = input('Masukkan harga sampah organik: ')
    non_organik = input('Masukkan harga sampah non organik: ')
    rating = 0

    if(str(nama_bank).replace(' ', '') == ''):
        print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
        kelolaBank()
    elif(str(lokasi).replace(' ', '') == ''):
        print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
        kelolaBank()
    elif(str(kontak).replace(' ', '') == ''):
        print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
        kelolaBank()
    elif(str(organik).replace(' ', '') == ''):
        print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
        kelolaBank()

    elif(str(non_organik).replace(' ', '') == ''):
        print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
        kelolaBank()
    elif(not re.match(r'[a-zA-Z]{3,$}', nama_bank)):
        print('\nNama bank sampah hanya boleh berisi huruf dan spasi! Minimal 3 huruf!')
        kelolaBank()
    elif(not re.match(r'^(?=(?:[^a-zA-Z]*[a-zA-Z]{3}){3})[a-zA-Z0-9\s,\.,\()]{10,}$', lokasi)):
        print('\nLokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurung, setidaknya memiliki 3 huruf! Minimal 10 karakter!')
        kelolaBank()
    elif(not re.match(r'^\d{7,15}$', kontak)):
        print('\nKontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!')
        kelolaBank()
    elif(not re.match(r'^\d{4,}$', organik or non_organik)):
        print('\nHarga sampah hanya boleh berisi angka, minimal 4 angka!')
        kelolaBank()
    elif(not re.match(r'^[1-9][0-9]*$', organik or non_organik)):
        print('\nHarga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
        kelolaBank()

    data = [[nama_bank, nama_pengelola, lokasi, f'{kontak}', organik, non_organik, rating]]

    new_data = pd.DataFrame(data, columns=header)
    new_data.to_csv('data_bank_sampah.csv', mode='a', sep=';', header=False, index=False)

    print('\nData bank sampah berhasil dibuat!')
    kelolaBank()

# Lihat data bank sampah pengelola
elif(choice == '2'):
    print('\n', bank_sampah_pengelola)
    kelolaBank()

# Edit Pengelola
elif(choice == '3'):
    print('\n', bank_sampah_pengelola)
    selected_bank = input('\nPilih bank sampah yang ingin di update berdasarkan nomor bank sampah (q jika tidak ada): ')

    if(selected_bank.replace(' ', '') == ''):
        print('\nInputan tidak boleh kosong!')
        kelolaBank()
    elif(not re.match(r'^\d+$', selected_bank)):
        print('\nInput hanya boleh berisi angka!')
        kelolaBank()
    elif(int(selected_bank) in bank_sampah_pengelola.index):
        data_dipilih = bank_sampah.iloc[int(selected_bank)]

        nama_bank = data_dipilih['nama bank sampah']
        nama_pengelola = data_dipilih['nama pengelola']
        lokasi = data_dipilih['lokasi']
        kontak = data_dipilih['kontak pengelola']
        harga_organik = data_dipilih['harga sampah organik']
        harga_non_organik = data_dipilih['harga sampah non-organik']
        rating = data_dipilih['rating']

        print('\nSilakan isi data terbaru, kosongkan jika tidak ada perubahan')
        new_nama_bank = input(f'Masukkan nama bank sampah baru ({nama_bank}): ') or nama_bank
        new_lokasi = input(f'Masukkan lokasi bank sampah baru ({lokasi}): ') or lokasi
        new_kontak = input(f'Masukkan kontak baru pengelola bank sampah ({kontak}): ') or kontak
        new_harga_organik = input(f'Masukkan harga baru sampah organik ({harga_organik}): ') or harga_organik
        new_harga_non_organik = input(f'Masukkan harga baru sampah non organik ({harga_non_organik}): ') or harga_non_organik

```

```

if(str(new_nama_bank).replace(' ', '') == ''):
    print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
    kelolaBank()
elif(str(new_lokasi).replace(' ', '') == ''):
    print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
    kelolaBank()
elif(str(new_kontak).replace(' ', '') == ''):
    print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
    kelolaBank()
elif(str(new_harga_organik).replace(' ', '') == ''):
    print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
    kelolaBank()
elif(str(new_harga_non_organik).replace(' ', '') == ''):
    print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
    kelolaBank()
elif(not re.match(r'^[a-zA-Z\s]{3,}$', new_nama_bank)):
    print('\nNama bank sampah dan nama pengelola hanya boleh berisi huruf dan spasi!
Minimal 3 huruf!')
    kelolaBank()
elif(not re.match(r'^(?:[^a-zA-Z]*[a-zA-Z]{3})[a-zA-Z0-9\s,\.()]{10,}$',
new_lokasi)):
    print('\nLokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurung,
setidaknya memiliki 3 huruf! Minimal 10 karakter!')
    kelolaBank()
elif(not re.match(r'^\d{7,15}$', str(new_kontak))):
    print('\nKontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15
angka!')
    kelolaBank()
elif(not re.match(r'^\d{4,}$', str(new_harga_organik))):
    print('\nHarga sampah organik hanya boleh berisi angka, minimal 4 angka!')
    kelolaBank()
elif(not re.match(r'^\d{4,}$', str(new_harga_non_organik))):
    print('\nHarga sampah non organik hanya boleh berisi angka, minimal 4 angka!')
    kelolaBank()

elif(not re.match(r'^[1-9][0-9]*$', str(new_harga_organik))):
    print('\nHarga sampah organik tidak boleh didahului angka 0! Harap masukkan harga
dengan benar!')
    kelolaBank()
elif(not re.match(r'^[1-9][0-9]*$', str(new_harga_non_organik))):
    print('\nHarga sampah non organik tidak boleh didahului angka 0! Harap masukkan harga
dengan benar!')
    kelolaBank()
|
bank_sampah.loc[int(selected_bank), ['nama bank sampah', 'nama pengelola', 'lokasi',
'kontak pengelola', 'harga sampah organik', 'harga sampah non-organik', 'rating']] = [new_nama_bank,
nama_pengelola, new_lokasi, int(f'{new_kontak}'), int(new_harga_organik), int(new_harga_non_organik),
rating]
bank_sampah.to_csv('data_bank_sampah.csv', index=False, sep=';')

print('\nData bank sampah Anda berhasil diperbarui!')
kelolaBank()
else:
    print('\nSilakan pilih bank sampah yang tersedia!')
    kelolaBank()

```

```

# Hapus data bank sampah pengelola
elif(choice == '4'):
    print('\n', bank_sampah_pengelola)
    delete_index = input('\nPilih nomor bank sampah yang ingin dihapus (q jika tidak ada): ')

    if(delete_index.lower() == 'q'):
        kelolaBank()
    elif(delete_index.replace(' ', '') == ''):
        print('\nInput tidak boleh kosong!')
        kelolaBank()
    elif(not re.match(r'^\d+$', delete_index)):
        print('\nInput hanya boleh berisi angka!')
        kelolaBank()
    elif(int(delete_index) in bank_sampah_pengelola.index):
        delete_bank_sampah = bank_sampah.drop(index=int(delete_index))
        delete_bank_sampah.to_csv('data_bank_sampah.csv', sep=';', index=False)

        print('\nData bank sampah berhasil dihapus!')
        kelolaBank()
    else:
        print('\nData bank sampah tidak ada dalam pilihan!')
        kelolaBank()

# Kembali
elif(choice == '5'):
    choicePengelola()

# Validasi angka <= 1 & >= 5
else:
    print('\nSilakan pilih opsi sesuai angka yang ada!')
    kelolaBank()

```

```

# Hapus data bank sampah pengelola
elif(choice == '4'):
    print('\n', bank_sampah_pengelola)
    delete_index = input('\nPilih nomor bank sampah yang ingin dihapus (q jika tidak ada): ')
    if(delete_index.lower() == 'q'):
        kelolaBank()
    elif(delete_index.replace(' ', '') == ''):
        print('\nInput tidak boleh kosong!')
        kelolaBank()
    elif(not re.match(r'^d+$', delete_index)):
        print('\nInput hanya boleh berisi angka!')
        kelolaBank()
    elif(int(delete_index) in bank_sampah_pengelola.index):
        delete_bank_sampah = bank_sampah.drop(index=int(delete_index))
        delete_bank_sampah.to_csv('data_bank_sampah.csv', sep=';', index=False)

        print('\nData bank sampah berhasil dihapus!')
        kelolaBank()
    else:
        print('\nData bank sampah tidak ada dalam pilihan!')
        kelolaBank()

# Kembali
elif(choice == '5'):
    choicePengelola()

# Validasi angka <= 1 & >= 5
else:
    print('\nSilakan pilih opsi sesuai angka yang ada!')
    kelolaBank()

if(auth['auth'] == 'admin'):
    choiceAdmin()
elif(auth['auth'] == 'user'):
    choiceUser()
else:
    user_rating = input(f'\nMasukkan rating untuk {bank_sampah["nama bank sampah"]}' + [int(choice)](1-5) (q untuk quit): ').strip()

    if(user_rating == 'q'):
        rating()
    elif(user_rating.replace(' ', '') == ''):
        print('\nRating tidak boleh kosong!')
        rating()
    elif(not re.match(r'^d{1}$', user_rating)):
        print('\nSilakan rating menggunakan angka yang sudah ditentukan!')
        rating()
    elif(user_rating == '1' or user_rating == '2' or user_rating == '3' or user_rating == '4' or user_rating == '5'):
        data = [[user_login['username'], choice, int(user_rating)]]

        new_data = pd.DataFrame(data, columns=header)
        new_data.to_csv('rating.csv', mode='a', sep=';', header=False, index=False)

        new_data_rating = pd.read_csv('rating.csv', sep=';')

        # simpan data rating di data bank sampah
        data_dipilih = bank_sampah.iloc[int(choice)]

        filtered_rating = new_data_rating[new_data_rating['id_bank'] == int(choice)]

        if(not filtered_rating.empty):
            avg_rating = filtered_rating['rating'].mean()
            f_avg_rating = f'{avg_rating:.1f}'

            bank_sampah.loc[int(choice), ['nama bank sampah', 'nama pengelola', 'lokasi', 'kontak pengelola', 'harga sampah organik', 'harga sampah non-organik', 'rating']] = [data_dipilih['nama bank sampah'], data_dipilih['nama pengelola'], data_dipilih['lokasi'], data_dipilih['kontak pengelola'], data_dipilih['harga sampah organik'], data_dipilih['harga sampah non-organik'], float(f_avg_rating)]
            bank_sampah.to_csv('data_bank_sampah.csv', index=False, sep=';')

            print(f'Terimakasih telah memberi penilaian untuk {bank_sampah["nama bank sampah"]}' + [int(choice)](1-5)!)
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()
            else:
                print('\nSilakan rating bank sampah dari 1 sampai 5!')
                rating()

        else:
            print('\nSilakan pilih opsi yang tersedia!')
            rating()

```

```

def search():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';', dtype={'kontak pengelola': str})

    print('\n' + 30 * '=')
    choice = input('\n[1]Nama Bank Sampah\n[2]Nama Pengelola Bank Sampah\n[3]Lokasi Bank
Sampah\n[4]Harga Sampah Organik\n[5]Harga Sampah Non Organik\n[6]Kembali\nCari Bank Sampah berdasarkan
kategori: ')

    if(choice == '1'):
        nama_bank = input('Masukkan nama bank sampah yang ingin dicari (q untuk quit): ')

        if(nama_bank == 'q'):
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()
            elif(auth['auth'] == 'pengelola'):
                choicePengelola()
        elif(nama_bank.replace(' ', '') == ''):
            print('\nInput pencarian tidak boleh kosong!')
            search()
        else:
            data_bank = bank_sampah[bank_sampah['nama bank sampah'].str.contains(nama_bank, case=False,
na=False)]

            if not data_bank.empty:
                print('\n')
                print(data_bank)
                search()
            else:
                print("\nTidak ada hasil yang mendekati nama bank sampah tersebut.")
                search()

    elif(choice == '2'):
        nama_pengelola = input('Masukkan nama pengelola bank sampah yang ingin dicari (q untuk quit):
')

        if(nama_pengelola == 'q'):
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()
            elif(auth['auth'] == 'pengelola'):
                choicePengelola()
        elif(nama_pengelola.replace(' ', '') == ''):
            print('\nInput pencarian tidak boleh kosong!')
            search()
        else:
            data_bank = bank_sampah[bank_sampah['nama pengelola'].str.contains(nama_pengelola,
case=False, na=False)]

            if not data_bank.empty:
                print('\n')
                print(data_bank)
                search()
            else:
                print("\nTidak ada hasil yang mendekati nama pengelola bank sampah tersebut.")
                search()

    elif(choice == '3'):
        lokasi = input('Masukkan lokasi bank sampah yang ingin dicari (q untuk quit): ')

        if(lokasi == 'q'):
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()

            elif(auth['auth'] == 'pengelola'):
                choicePengelola()
        elif(lokasi.replace(' ', '') == ''):
            print('\nInput pencarian tidak boleh kosong!')
            search()
        else:
            data_bank = bank_sampah[bank_sampah['lokasi'].str.contains(lokasi, case=False, na=False)]

            if not data_bank.empty:
                print('\n')
                print(data_bank)
                search()
            else:
                print("\nTidak ada hasil yang mendekati lokasi bank sampah tersebut.")
                search()

    elif(choice == '4'):
        harga_organik = input('Masukkan harga sampah organik yang ingin dicari (q untuk quit): ')

        if(harga_organik == 'q'):
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()
            elif(auth['auth'] == 'pengelola'):
                choicePengelola()
        elif(harga_organik.replace(' ', '') == ''):
            print('\nInput pencarian tidak boleh kosong!')
            search()
        elif(not re.match(r'^\d+$', harga_organik)):
            print('\nPencarian harga organik hanya boleh menginputkan angka saja!')
            search()
        else:
            data_bank = bank_sampah[bank_sampah['harga sampah organik'] <= int(harga_organik)]

```

```

        if not data_bank.empty:
            print('\n')
            print(data_bank)
            search()
        else:
            print("\nTidak ada hasil yang mendekati harga sampah tersebut.")
            search()
    elif(choice == '5'):
        harga_non_organik = input('Masukkan harga sampah non organik yang ingin dicari (q untuk quit):')

    if(harga_non_organik == 'q'):
        if(auth['auth'] == 'admin'):
            choiceAdmin()
        elif(auth['auth'] == 'user'):
            choiceUser()
        elif(auth['auth'] == 'pengelola'):
            choicePengelola()
        elif(harga_non_organik.replace(' ', '') == ''):
            print('\nInput pencarian tidak boleh kosong!')
            search()
        elif(not re.match(r'^\d+$', harga_non_organik)):
            print('\nPencarian harga non organik hanya boleh menginputkan angka saja!')
            search()
        else:
            data_bank = bank_sampah[bank_sampah['harga sampah non-organik'] <= int(harga_non_organik)]

            if not data_bank.empty:
                print('\n')
                print(data_bank)
                search()
            else:
                print("\nTidak ada hasil yang mendekati harga sampah tersebut.")
                search()

    else:
        print("\nTidak ada hasil yang mendekati harga sampah tersebut.")
        search()

```

```

    elif(choice == '6'):
        if(auth['auth'] == 'admin'):
            choiceAdmin()
        elif(auth['auth'] == 'user'):
            choiceUser()
        elif(auth['auth'] == 'pengelola'):
            choicePengelola()
        else:
            print('\nSilakan pilih kategori berdasarkan angka yang tersedia!')
            search()

```

```

def crud():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';', dtype={'kontak pengelola': str})
    header = ['nama bank sampah', 'nama pengelola', 'lokasi', 'kontak pengelola', 'harga sampah organik', 'harga sampah non-organik', 'rating']

    print('\n' + 30 * '=')
    choice = input('\n[1]Tambah data bank sampah\n[2]Lihat data bank sampah\n[3]Update data bank sampah\n[4]Hapus data bank sampah\n[5]Kembali\nSilakan pilih fitur berdasarkan angka: ')

    if(choice == '1'):
        nama_bank = input('\nMasukan nama bank sampah: ')
        nama_pengelola = input('Masukkan nama pengelola: ')
        lokasi = input('Masukkan lokasi bank sampah: ')
        kontak = input('Masukkan kontak pengelola bank sampah: ')
        organik = input('Masukkan harga sampah organik: ')
        non_organik = input('Masukkan harga sampah non organik: ')
        rating = 0

        # regex docs : https://docs.python.org/3/howto/regex.html

        if(str(nama_bank).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(nama_pengelola).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(lokasi).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(kontak).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()

```

```

        elif(str(organik).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(non_organik).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(not re.match(r'^[a-zA-Z\s]{3,20$', nama_bank)):
            print('\nNama bank sampah hanya boleh berisi huruf dan spasi! Minimal 3 huruf, maximal 20
huruf!')
            crud()
        elif(not re.match(r'^[a-zA-Z\s]{3,20$', nama_pengelola)):
            print('\nNama pengelola hanya boleh berisi huruf dan spasi! Minimal 3 huruf!, maximal 20
huruf')
            crud()
        elif(not re.match(r'^(?=[^a-zA-Z]*[a-zA-Z])[3]{a-zA-Z0-9\s,\.()}{10,255}$', lokasi)):
            print('\nLokasi hanya boleh berisi huruf, spasi, angka, koma dan tanda kurung, setidaknya
memiliki 3 huruf! Minimal 10 karakter!')
            crud()
        elif(not re.match(r'^\d{7,15}$', kontak)):
            print('\nKontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!')
            crud()
        elif(not re.match(r'^\d{4,7}$', organik)):
            print('\nHarga sampah hanya boleh berisi angka, minimal 4 angka!')
            crud()
        elif(not re.match(r'^\d{4,7}$', non_organik)):
            print('\nHarga sampah hanya boleh berisi angka, minimal 4 angka!')
            crud()
        elif(not re.match(r'^[1-9][0-9]*$', organik)):
            print('\nHarga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
            crud()
        elif(not re.match(r'^[1-9][0-9]*$', non_organik)):
            print('\nHarga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
            crud()

data = [[nama_bank, nama_pengelola, lokasi, f'{kontak}', organik, non_organik, rating]]

new_data = pd.DataFrame(data, columns=header)
new_data.to_csv('data_bank_sampah.csv', mode='a', sep=';', header=False, index=False)

print('\nData bank sampah berhasil dibuat!!!')
crud()
elif(choice == '2'):
    print(bank_sampah)
    crud()
elif(choice == '3'):
    print('\n')
    print(bank_sampah)
    selected_bank = input('\nPilih bank sampah yang ingin di update berdasarkan nomor bank sampah
(q jika tidak ada): ')

    if(selected_bank.lower() == 'q'):
        crud()
    elif(selected_bank.replace(' ', '') == ''):
        print('\nNomor bank sampah tidak boleh kosong!')
        crud()
    elif(not re.match(r'^\d+$', selected_bank)):
        print('\nSilakan isi menggunakan angka!')
        crud()
    elif(int(selected_bank) in bank_sampah.index):
        data_dipilih = bank_sampah.iloc[int(selected_bank)]

        nama_bank = data_dipilih['nama bank sampah']
        nama_pengelola = data_dipilih['nama pengelola']
        lokasi = data_dipilih['lokasi']
        kontak = data_dipilih['kontak pengelola']
        harga_organik = data_dipilih['harga sampah organik']
        harga_non_organik = data_dipilih['harga sampah non-organik']
        rating = data_dipilih['rating']

```

```

        print('\nSilakan isi data terbaru, kosongkan jika tidak ada perubahan')
        new_nama_bank = input(f'Masukkan nama bank sampah baru ({nama_bank}): ') or nama_bank
        new_nama_pengelola = input(f'Masukkan nama pengelola baru ({nama_pengelola}): ') or
        nama_pengelola
        new_lokasi = input(f'Masukkan lokasi bank sampah baru ({lokasi}): ') or lokasi
        new_kontak = input(f'Masukkan kontak baru pengelola bank sampah ({kontak}): ') or kontak
        new_harga_organik = input(f'Masukkan harga baru sampah organik ({harga_organik}): ') or
        harga_organik
        new_harga_non_organik = input(f'Masukkan harga baru sampah non organik
        ({harga_non_organik}): ') or harga_non_organik

        if(str(new_nama_bank).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        if(str(new_nama_pengelola).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(new_lokasi).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(new_kontak).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(new_harga_organik).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(str(new_harga_non_organik).replace(' ', '') == ''):
            print('\nInput tidak boleh kosong atau spasi saja! Mohon input data dengan benar!')
            kelolaBank()
        elif(not re.match(r'^[a-zA-Z]{3,20}$', new_nama_bank)):
            print('\nNama bank sampah hanya boleh berisi huruf dan spasi! Minimal 3 huruf, maximal
            20 huruf!')

            crud()
        elif(not re.match(r'^[a-zA-Z\s]{3,20}$', new_nama_pengelola)):
            print('\nNama pengelola hanya boleh berisi huruf dan spasi! Minimal 3 huruf, maximal 20
            huruf!')
            crud()
        elif(not re.match(r'^(?:[^a-zA-Z]*[a-zA-Z]{3})[a-zA-Z0-9\s,\.()]{10,255}$',
        new_lokasi)):
            print('\nLokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurung,
            setidaknya memiliki 3 huruf! Minimal 10 karakter!')
            crud()
        elif(not re.match(r'^\d{7,15}$', str(new_kontak))):
            print('\nKontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15
            angka!')
            crud()
        elif(not re.match(r'^\d{4,7}$', str(new_harga_organik))):
            print('\nHarga sampah hanya boleh berisi angka, minimal 4 angka!')
            crud()
        elif(not re.match(r'^\d{4,7}$', str(new_harga_non_organik))):
            print('\nHarga sampah hanya boleh berisi angka, minimal 4 angka!')
            crud()
        elif(not re.match(r'^[1-9][0-9]*$', str(new_harga_organik))):
            print('\nHarga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan
            benar!')
            crud()
        elif(not re.match(r'^[1-9][0-9]*$', str(new_harga_non_organik))):
            print('\nHarga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan
            benar!')
            crud()

        bank_sampah.loc[int(selected_bank), ['nama bank sampah', 'nama pengelola', 'lokasi',
        'kontak pengelola', 'harga sampah organik', 'harga sampah non-organik', 'rating']] = [new_nama_bank,
        new_nama_pengelola, new_lokasi, new_kontak, new_harga_organik, new_harga_non_organik, rating]

        bank_sampah.to_csv('data_bank_sampah.csv', index=False, sep=';')

        print('\nData bank sampah berhasil diperbarui!!!!')
        crud()
    else:
        print('\nData belum ada, silakan masukkan data dengan benar!')
        crud()
    elif(choice == '4'):
        print('\n')
        print(bank_sampah)
        print('\n')
        delete_index = input('Pilih nomor bank sampah yang ingin dihapus (q jika tidak ada): ')

        if(delete_index.lower() == 'q'):
            crud()
        elif(delete_index.replace(' ', '') == ''):
            print('Nomor bank sampah tidak boleh kosong!')
            crud()
        elif(not re.match(r'^\d$', delete_index)):
            print('Silakan isi menggunakan angka!')
            crud()
        elif(int(delete_index) in bank_sampah):
            delete_bank_sampah = bank_sampah.drop(index=int(delete_index))
            delete_bank_sampah.to_csv('data_bank_sampah.csv', sep=';', index=False)
            print('\n' + 'Data bank sampah berhasil dihapus!!!!')
            crud()
        else:
            print('Tidak ada data bank sampah! Silakan pilih nomor bank sampah yang sesuai!')
            crud()
    elif(choice == '5'):
        if(auth['auth'] == 'admin'):
            choiceAdmin()
        elif(auth['auth'] == 'pengelola'):
            choicePengelola()

```

```

def ecoscalc():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';', dtype={'kontak pengelola': str})

    print('\n' + 30 * '=')
    print(bank_sampah)
    selected_bank = input('\nPilih bank sampah berdasarkan nomor bank sampah (q jika tidak ada): ')

    if(selected_bank.lower() == 'q'):
        if(auth['auth'] == 'admin'):
            choiceAdmin()
        elif(auth['auth'] == 'user'):
            choiceUser()
        elif(auth['auth'] == 'pengelola'):
            choicePengelola()
    elif(selected_bank.replace(' ', '') == ""):
        print('Nomor bank sampah tidak boleh kosong!')
        ecoscalc()
    elif(not re.match(r'^\d+$', selected_bank)):
        print('\nSilakan pilih bank sampah menggunakan angka!')
        ecoscalc()
    elif(int(selected_bank) in bank_sampah.index):
        data_dipilih = bank_sampah.iloc[int(selected_bank)] 

        harga_organik = data_dipilih['harga sampah organik']
        harga_non_organik = data_dipilih['harga sampah non-organik']

        input_organik = input('\nMasukkan berat sampah organik(kg): ')
        input_non_organik = input('Masukkan berat sampah non-organik(kg): ')

        if(input_organik.replace(' ', '') == ""):
            print('\nBerat sampah organik tidak boleh kosong!')
            ecoscalc()
        elif(input_non_organik.replace(' ', '') == ""):
            print('\nBerat sampah non organik tidak boleh kosong!')

        ecoscalc()
        elif(not re.match(r'^\d{1,3}$', input_organik)):
            print('\nHarga sampah organik hanya boleh berisi angka! Maximal 3 digit!')
            ecoscalc()
        elif(not re.match(r'^\d{1,3}$', input_non_organik)):
            print('\nHarga sampah non organik hanya boleh berisi angka! Maximal 3 digit!')
            ecoscalc()
        elif(not re.match(r'^[1-9][0-9]*$', input_organik)):
            print('Harga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
            ecoscalc()
        elif(not re.match(r'^[1-9][0-9]*$', input_non_organik)):
            print('Harga sampah tidak boleh didahului angka 0! Harap masukkan harga dengan benar!')
            ecoscalc()

        print(f'\nTotal harga sampah organik anda adalah Rp.{int(harga_organik) * 
int(input_organik):,.0f}')
        print(f'Total harga sampah non-organik anda adalah Rp.{int(harga_non_organik) * 
int(input_non_organik):,.0f}')

        next_step = input('\n[1]Ya\n[2]Tidak\nApakah anda ingin menghitung lagi? ')

        if(next_step.replace(' ', '') == ""):
            print('\nInput tidak boleh kosong!')
            ecoscalc()
        elif(next_step == '1'):
            ecoscalc()
        elif(next_step == '2'):
            if(auth['auth'] == 'admin'):
                choiceAdmin()
            elif(auth['auth'] == 'user'):
                choiceUser()
            elif(auth['auth'] == 'pengelola'):
                choicePengelola()
        else:
            print('\nSilakan pilih berdasarkan pilihan yang tersedia!')

    else:
        print('\nData bank sampah belum ada, silakan masukkan data dengan benar!')
        ecoscalc()

```

```

# ! CHOICE START

def choiceAdmin():
    print('\n' + 30 * '=')
    choice = input('\n[1]EcosCalc\n[2]CRUD Bank Sampah\n[3]Search Bank Sampah\n[4]Rating Bank
Sampah\n[5]Logout\n[6]Exit\nSilakan pilih fitur berdasarkan angka: ')
    if(choice == '1' or choice == '2' or choice == '3' or choice == '4' or choice == '5' or choice ==
'6'):
        if(int(choice) == 1):
            ecoscalc()
        elif(int(choice) == 2):
            crud()
        elif(int(choice) == 3):
            search()
        elif(int(choice) == 4):
            rating()
        elif(int(choice) == 5):
            print('\nLogout berhasil!!!')
            login()
        elif(int(choice) == 6):
            print('\n' + 30 * '=')
            print('\nTerimakasih telah menggunakan EcoSphera!\n')
            exit()
    else:
        print('\nSilakan pilih berdasarkan angka atau pilihan yang tersedia')
        choiceAdmin()

def choiceUser():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';')

    print('\n' + 30 * '=')
    choice = input('\n[1]EcosCalc\n[2]Lihat Bank Sampah\n[3]Search Bank Sampah\n[4]Rating Bank
Sampah\n[5]Logout\n[6]Exit\nSilakan pilih fitur berdasarkan angka: ')

    if(choice == '1' or choice == '2' or choice == '3' or choice == '4' or choice == '5' or choice ==
'6'):
        if(int(choice) == 1):
            ecoscalc()
        elif(int(choice) == 2):
            print('\n', bank_sampah)
            choiceUser()
        elif(int(choice) == 3):
            search()
        elif(int(choice) == 4):
            rating()
        elif(int(choice) == 5):
            login()
        elif(int(choice) == 6):
            print('\n' + 30 * '=')
            print('\nTerimakasih telah menggunakan EcoSphera!\n')
            exit()
    else:
        print('\nSilakan pilih berdasarkan angka atau pilihan yang tersedia')
        choiceUser()

def choicePengelola():
    bank_sampah = pd.read_csv('data_bank_sampah.csv', sep=';')

    print('\n' + 30 * '=')
    choice = input('\n[1]EcosCalc\n[2]Kelola Bank Sampah\n[3]Lihat Bank Sampah\n[4]Search Bank
Sampah\n[5]Logout\n[6]Exit\nSilakan pilih fitur berdasarkan angka: ')

    if(choice == '1' or choice == '2' or choice == '3' or choice == '4' or choice == '5' or choice ==
'6'):
        if(int(choice) == 1):
            ecoscalc()
        elif(int(choice) == 2):
            kelolaBank()
        elif(int(choice) == 3):
            print('\n', bank_sampah)
            choicePengelola()
        elif(int(choice) == 4):
            search()
        elif(int(choice) == 5):
            print('\nLogout berhasil!!!')
            login()
        elif(int(choice) == 6):
            print('\n' + 30 * '=')
            print('\nTerimakasih telah menggunakan EcoSphera!\n')
            exit()
    else:
        print('\nSilakan pilih berdasarkan angka atau pilihan yang tersedia')
        choiceAdmin()

# ! CHOICE END

chance = {'chance': 2}

```

```

def login():
    users = pd.read_csv('users.csv', sep=';')

    print('\n' + 30 * '=')

    print('\nSilakan masuk ke akun anda!\n')
    uname = input('Masukkan username: ')
    pw = input('Masukkan password: ')

    validate = users[(users['username'] == uname) & (users['password'] == pw)]

    if(chance['chance'] == 0):
        chance['chance'] = 2
        choice1()
    elif(uname.replace(' ', '') == '' or pw.replace(' ', '') == ''):
        print(f'\nUsername atau password tidak boleh kosong! Silakan coba lagi! Kesempatan anda {chance["chance"]}x lagi!')
        chance['chance'] -= 1
        login()

    else:
        if(not validate.empty):
            print('\n' + 30 * '=')
            print('\nLogin berhasil!!!')

            chance['chance'] = 2
            role = validate.iloc[0]['role']

            user_login['username'] = uname

            if(role == 'admin'):
                auth['auth'] = 'admin'
                choiceAdmin()
            elif(role == 'user'):
                auth['auth'] = 'user'
                choiceUser()
            elif(role == 'pengelola'):
                auth['auth'] = 'pengelola'
                choicePengelola()
            else:
                print('laahh siapa dong ini')
            else:
                print(f'\nUsername atau password salah! Silakan coba lagi! Kesempatan anda {chance["chance"]}x lagi!')
                chance['chance'] -= 1
                login()
        else:
            print('')

def login():
    users = pd.read_csv('users.csv', sep=';')

    print('\n' + 30 * '=')

    print('\nSilakan masuk ke akun anda!\n')
    uname = input('Masukkan username: ')
    pw = input('Masukkan password: ')

    validate = users[(users['username'] == uname) & (users['password'] == pw)]

    if(chance['chance'] == 0):
        chance['chance'] = 2
        choice1()
    elif(uname.replace(' ', '') == '' or pw.replace(' ', '') == ''):
        print(f'\nUsername atau password tidak boleh kosong! Silakan coba lagi! Kesempatan anda {chance["chance"]}x lagi!')
        users = pd.read_csv('users.csv', sep=';')

    print('\n' + 30 * '=')

    regisChoice = input('\n[1]User\n[2]Pengelola\n[3]Kembali\n[4]Exit\nSilakan pilih register sebagai:')

    if(regisChoice == '1' or regisChoice == '2'):
        def regis1():
            uname = input('\nMasukkan username: ')
            pw = input('Masukkan password: ')

            notUnique = users[(users['username'] == uname)]
            if(chance['chance'] == 0):
                chance['chance'] = 2
                choice1()
            elif(not notUnique.empty):
                print('\nUsername sudah terdaftar! Silakan pakai username baru!')

```

```

        chance['chance'] -= 1
        regis2()
    elif(uname.replace(' ', '') == '' or pw.replace(' ', '') == ''):
        print('\nUsername atau password tidak boleh kosong!')
        chance['chance'] -= 1
        regis2()
    elif(not re.match(r'^\w{3,20}$', uname)):
        print('\nUsername hanya boleh mengandung huruf, angka dan garis bawah! Minimal 3 karakter, Maximal 20 karakter!')
        chance['chance'] -= 1
        regis2()
    elif(not re.match(r'^(?=.*[a-z])(?=.*[A-Z])(?=.*\d){8,}', pw)):
        print('Password setidaknya mengandung huruf besar, huruf kecil dan angka! Minimal 8 karakter!')
        chance['chance'] -= 1
        regis2()
    else:
        header = ['username', 'password', 'role']

        if(int(regisChoice) == 1):
            user = [[uname, pw, 'user']]
        elif(int(regisChoice) == 2):
            user = [[uname, pw, 'pengelola']]

        new_user = pd.DataFrame(user, columns=header)
        new_user.to_csv('users.csv', mode='a', sep=';', header=False, index=False)

        print('\nSelamat akun anda telah berhasil dibuat!!! Silakan login terlebih dahulu!')
        chance['chance'] = 2
        login()

    regis2()
    elif(regisChoice == '3'):
        choice1()
    elif(regisChoice == '4'):
        print('\nTerimakasih telah menggunakan EcoSphera!\n')
        exit()
    else:
        print('\nSilakan pilih berdasarkan angka yang tersedia!!!')
        register()

def choice1():
    print('\n' + 30 * '=')
    print('Selamat Datang di EcoSphera!!!')

    choice = input('\n[1]Register\n[2]Login\n[3]Exit\nSilakan pilih Login atau Register: ')

    if(choice == '1' or choice == '2' or choice == '3'):
        if(int(choice) == 1):
            register()
        elif(int(choice) == 2):
            login()
        elif(int(choice) == 3):
            print('\nTerimakasih telah menggunakan EcoSphera!\n')
            exit()
    else:
        print('\nSilakan pilih opsi yang tersedia!')
        choice1()

    choice1()ng! Silakan coba lagi! Kesempatan anda {chance['chance']}x lagi!
    chance['chance'] -= 1
    login()

choice1()ng! Silakan coba lagi! Kesempatan anda {chance['chance']}x lagi!
    chance['chance'] -= 1
    login()
else:
    if(not validate.empty()):
        print('\n' + 30 * '=')
        print('\nLogin berhasil!!!')

        chance['chance'] = 2
        role = validate.iloc[0]['role']

        user_login['username'] = uname

        if(role == 'admin'):
            auth['auth'] = 'admin'
            choiceAdmin()
        elif(role == 'user'):
            auth['auth'] = 'user'
            choiceUser()
        elif(role == 'pengelola'):
            auth['auth'] = 'pengelola'
            choicePengelola()
        else:
            print('laahh siapa dong ini')

        else:
            print(f'\nUsername atau password salah! Silakan coba lagi! Kesempatan anda {chance['chance']}x lagi!')
            chance['chance'] -= 1
            login()

```

## **11. Pengujian**

### **11.1 Lingkungan Pengujian**

#### **11.1.1 Perangkat Lunak Pengujian**

Pengujian EcoSphera dijalankan dalam CLI dengan menggunakan perangkat lunak sebagai berikut.

- Sistem Operasi: Windows 11
- Bahasa Pemrograman: Python
- Database: CSV

#### **11.1.2 Perangkat Keras Pengujian**

Perangkat keras yang digunakan dalam melakukan pengujian EcoSphera yaitu laptop Lenovo X390 dengan spesifikasi berikut.

- Processor: Intel(R) Core(TM) i5-8265U
- Memory: 16GB
- Storage: 512GB

#### **11.1.3 Material Pengujian**

Menggunakan website EcoSPhera, pengelola bank sampah dapat membuat, menambahkan, update data bank sampah, dan menghapus data bank sampah yang dikelola. Data bank sampah tersebut diantaranya, nama bank sampah, lokasi, kontak pengelola, serta harga bank sampah berdasarkan jenis sampah organik dan non-organik. Pengelola juga dapat melihat data rating yang diberikan oleh user untuk bank sampah yang dikelolanya. Melalui EcoSphera, user dapat melihat lokasi bank sampah melalui fitur pencarian juga informasi detail mengenai bank sampah yang dicarinya. Selain itu user dapat memberikan rating bank sampah untuk pengelola.

#### **11.1.4 Sumber Daya Manusia**

Sumber daya manusia pada pengujian website EcoSphera melibatkan lima orang, terdiri atas empat anggota kelompok dan satu orang pengujil dari luar kelompok. Penguji merupakan mahasiswa program studi Rekayasa Perangkat Lunak kampus UPI di Cibiru yang telah mempelajari dan memahami bahasa pemrograman python. Tujuan dari penguji untuk mencari kekurangan atau kesalahan dan mengevaluasi program yang dikembangkan.

#### **11.1.5 Prosedur Umum Pengujian**

- Pengenalan dan Latihan

Penguji diberikan pengenalan mengenai program yang dibuat mengenai tujuan dan fitur-fitur yang disediakan oleh EcoSphera. Mengenai perangkat lunak yang digunakan, penguji pada dasarnya sudah paham mengenai pengujian yang harus dilakukan. Latihan ini bertujuan untuk memastikan pemahaman terhadap teknis pengujian EcoSphera.

- **Persiapan Perangkat Keras**

Persiapan perangkat keras berupa laptop yang akan digunakan untuk pengujian.

- **Persipan Perangkat Lunak**

Perangkat lunak pendukung yang digunakan dalam pengujian yaitu.

- a. Instalasi Visual Studio Code versi 1.96.2

- b. Terminal

- **Pelaksanaan Pengujian**

Pelaksanaan dilakukan dengan melakukan pengujian terhadap code pada program EcoSphera berdasarkan test case yang telah disusun oleh anggota kelompok (developer). Test case ini mencakup berbagai skenario pengujian untuk memastikan bahwa semua fitur dalam aplikasi berfungsi sesuai spesifikasi. Pengujian yang dilakukan di antaranya, menjalankan fitur register dan login, menjalankan fungsi pencarian bank sampah, mencoba fitur CRUD data bank sampah, melakukan testing untuk pemberian rating bank sampah, dan memasukan berat sampah untuk menghitung total berdasarkan jenis sampah. Hasil pengujian untuk setiap test case dicatat langsung ke dalam dokumen test case, mencakup kesesuaian hasil dengan harapan, baik itu sesuai atau tidak sesuai. Setelah pengembang menyelesaikan perbaikan berdasarkan catatan di test case, dilakukan kembali pengujian yang sama untuk memastikan masalah telah terselesaikan. Dengan pengisian secara langsung hasil pengujian pada dokumen test case, proses pengujian menjadi lebih terstruktur. Hal ini dimaksudkan agar setiap langkah pengujian dapat dievaluasi dan ditelusuri dengan mudah.

- **Pelaporan Hasil**

Pelaporan hasil pengujian EcoSphera disusun berdasarkan data yang telah dicatat selama pelaksanaan pengujian, yang terdokumentasi dalam dokumen test case. Test case yang telah diisi sesuai dengan hasil pengujian akan dimasukkan ke dalam dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL). Dokumen ini kemudian diserahkan kepada dosen mata kuliah Program Studi Rekayasa Perangkat Lunak untuk penilaian lebih lanjut.

## 11.2 Hasil Pengujian

### Test Case Login

TestCase ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	PASS/FAIL
LG01	Login dengan data valid sebagai admin	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: admin Password: admin	Login Berhasil dan masuk menu homepage admin	Sesuai dengan ekspektasi	Valid
LG02	Login dengan data valid sebagai pengelola	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: pengelola Password: pengelola	Login Berhasil dan masuk menu home pengelola	Sesuai dengan ekspektasi	Valid
LG03	Login dengan data valid sebagai pengguna	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: rafi Password: raf1	Login Berhasil dan masuk menu home pengguna	Sesuai dengan ekspektasi	Valid
LG04	Login dengan data invalid sebagai admin	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: admin Password: admin	Menampilkan peringatan username/pasword salah dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG05	Login dengan data invalid sebagai pengelola	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: pengelola Password: mnckcjkd	Menampilkan peringatan username/pasword salah dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG06	Login dengan data invalid sebagai pengguna	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter	Username: rafi Password: admunjbdhkc b	Menampilkan peringatan username/pasword salah dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG07	Login dengan 3 kali kesalahan	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter 6. Ulangi 3 kali	Username: rafi Password: admunjbdhkc b	Kembali ke menu awal dan menampilkan notifikasi kesempatan login habis	Sesuai dengan ekspektasi	Valid
LG10	Login pada Percobaan Kedua	1. Jalankan Program 2. Ketik 2 3. Masukkan Invalid Username 4. Masukkan Invalid Password 5. Klik Enter 6. Masukkan Valid Username 7. Masukkan Valid Password 8. Klik Enter	Invalid Username: Jaki Invalid Password: Jak18 Valid Username: rafi Valid Password: raf1	Login Berhasil dan masuk menu home	Sesuai dengan ekspektasi	Valid
LG11	Login dengan Username kosong dan password valid	1. Jalankan Program 2. Ketik 2 3. Kosongkan Username 4. Masukkan Password 5. Klik Enter	Username: Password: user	Menampilkan peringatan username/pasword tidak boleh kosong dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG12	Login dengan Username valid dan password kosong	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Kosongkan Password 5. Klik Enter	Username: user Password:	Menampilkan peringatan username/pasword tidak boleh kosong dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG13	Login dengan Username kosong dan password kosong	1. Jalankan Program 2. Ketik 2 3. Kosongkan Username 4. Kosongkan Password 5. Klik Enter	Username: Password:	Menampilkan peringatan username/pasword tidak boleh kosong dan pengurangan kesempatan login	Sesuai dengan ekspektasi	Valid
LG14	Login setelah Logout	1. Jalankan Program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Klik Enter 6. Ketik 6 7. Klik Enter 8. Jalankan Program 9. Ketik 2 10. Masukkan Username 11. Masukkan Password 12. Klik Enter	Username: user Password: user	Login Berhasil dan masuk menu home	Sesuai dengan ekspektasi	Valid

## Test Case Register

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
RG01	Register menggunakan data valid	1. Jalankan program 2. Ketik 1 3. ketik 1 4. Masukkan Username 5. Klik enter 6. Masukkan Password 8. Klik enter	1. Username: admin 2. Password: admin	Data pengguna tersimpan, redirect ke halaman pilihan login	Sesuai dengan ekspektasi	Valid
RG02	Register menggunakan data kosong	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Klik enter 5. Klik enter	1. Username: 2. Password:	Data tidak tersimpan, menampilkan return alert message username dan password tidak boleh kosong dan kesempatan register berkurang	Sesuai dengan ekspektasi	Valid
RG03	Register menggunakan username kosong namun password di isi	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Masukkan Password 6. Klik enter	1. Username: 2. Password: admin	Data tidak tersimpan, menampilkan return alert message username dan password tidak boleh kosong dan kesempatan register berkurang	Sesuai dengan ekspektasi	Valid
RG04	Register menggunakan symbol	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Masukkan Username 6. Klik enter 7. Masukkan Password 8. Klik Enter	1. Username: @#\$@@ 2. Password: \$@%\$@	Data tidak tersimpan, menampilkan return alert message username dan password hanya boleh menggunakan huruf, angka dan garis bawah	Sesuai dengan ekspektasi	Valid
RG05	Logout	1. Jalankan program 2. Ketik 2 3. Klik enter 4. Masukkan Username 6. Klik enter 7. Masukkan Password 8. Klik Enter 9. Ketik 5 10. Klik Enter	1. Username: admin 2. Password: admin	Kembali ke halaman login	Sesuai dengan ekspektasi	Valid
RG06	Register menggunakan spasi	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Masukkan Username 6. Klik enter 7. Masukkan Password 8. Klik Enter	1. Username: admin 1 2. Password: admin 1	Data tidak tersimpan, menampilkan return alert message username dan password hanya boleh menggunakan huruf, angka dan garis bawah	Sesuai dengan ekspektasi	Valid
RG07	Register dengan username lebih dari 256 karakter	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Masukkan Username 6. Klik enter 7. Masukkan Password 8. Klik Enter	1. Username: a(256x) 2. Password: admin1	Data tidak tersimpan, menampilkan error message username melebihi 20 karakter	Sesuai dengan ekspektasi	Valid
RG08	Register dengan username atau password mengandung unicode	1. Jalankan program 2. Ketik 1 3. Klik enter 4. Masukkan Username 6. Klik enter 7. Masukkan Password 8. Klik Enter	1. Username: user 你好 2. Password: user 你好	Data tidak tersimpan, menampilkan return alert message username dan password hanya boleh menggunakan huruf, angka dan garis bawah	Sesuai dengan ekspektasi	Valid

## Test Case Create

TestCase ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	PASS/FAIL
TC01	Create bank sampah sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: 15000	Muncul pemberitahuan data bank sampah berhasil dibuat dan kembali ke menu home admin	Sesuai dengan ekspektasi	Valid
TC02	Create bank sampah sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: 20000 Harga sampah non organik: 30000	Muncul pemberitahuan data bank sampah berhasil dibuat dan kembali ke menu home pengelola	Sesuai dengan ekspektasi	Valid
TC03	Create bank sampah dengan data kosong sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Kosongkan nama bank sampah 8. Kosongkan nama pengelola 9. Kosongkan Lokasi 10. Kosongkan kontak pengelola 11. Kosongkan harga sampah organik 12. Kosongkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: - Nama Pengelola: - Lokasi: - Kontak Pengelola: - Harga sampah organik: - Harga sampah non organik: -	Muncul Pemberitahuan bahwa data bank sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC04	Create bank sampah dengan data kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Kosongkan nama bank sampah 9. Kosongkan Lokasi 10. Kosongkan kontak pengelola 11. Kosongkan harga sampah organik 12. Kosongkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Lokasi: Kontak Pengelola: Harga sampah organik: Harga sampah non organik:	Muncul Pemberitahuan bahwa data bank sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC05	Create bank sampah dengan format penulisan kontak menggunakan string sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: Kosong Delapan Harga sampah organik: 10000 Harga sampah non organik: 15000	Muncul Pemberitahuan bahwa kontak harus diisi dengan angka minimal 7 dan maksimal 15	Sesuai dengan ekspektasi	Valid

TC06	Create bank sampah dengan format penulisan kontak menggunakan string sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: Kosong delapan Harga sampah organik: 20000 Harga sampah non organik: 30000	Muncul Pemberitahuan bahwa kontak harus diisi dengan angka minimal 7 dan maksimal 15	Sesuai dengan ekspektasi	Valid
TC07	Create bank sampah dengan format penulisan harga sampah organik menggunakan string sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: Sepuluh Harga sampah non organik: 15000	Muncul Pemberitahuan bahwa harga sampah harus diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC08	Create bank sampah dengan format penulisan harga sampah organik menggunakan string sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: Dua Harga sampah non organik: 30000	Muncul Pemberitahuan bahwa harga sampah harus diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC09	Create bank sampah dengan format penulisan harga sampah non-organik menggunakan string sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: Lima Belas Ribu	Muncul Pemberitahuan bahwa harga sampah harus diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC10	Create bank sampah dengan format penulisan harga sampah non-organik menggunakan string sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: 20000 Harga sampah non organik: Tiga Puluh Ribu	Muncul Pemberitahuan bahwa harga sampah harus diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC11	Create bank sampah dengan format penulisan nama bank sampah menggunakan simbol sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: <>#%\$*\$ Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: 15000	Muncul Pemberitahuan bahwa nama bank sampah tidak mendukung simbol	Sesuai dengan ekspektasi	Valid
TC12	Create bank sampah dengan format penulisan nama bank sampah menggunakan simbol sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: <>#%&*& Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: 20000 Harga sampah non organik: 30000	Muncul Pemberitahuan bahwa nama bank sampah tidak mendukung simbol	Sesuai dengan ekspektasi	Valid

TC13	Create bank sampah dengan data kontak kosong sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: - Harga sampah organik: 10000 Harga sampah non organik: 15000	Muncul pemberitahuan bahwa data kontak tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC14	Create bank sampah dengan data kontak kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: - Harga sampah organik: 20000 Harga sampah non organik: 30000	Muncul pemberitahuan bahwa data kontak tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC15	Create bank sampah dengan data harga sampah organik kosong sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: - Harga sampah non organik: 15000	Muncul pemberitahuan bahwa harga sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC16	Create bank sampah dengan data harga sampah organik kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: - Harga sampah non organik: 30000	Muncul pemberitahuan bahwa harga sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC17	Create bank sampah dengan data harga sampah non-organik kosong sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: -	Muncul pemberitahuan bahwa harga sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC18	Create bank sampah dengan data harga sampah non-organik kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Village Kontak Pengelola: 08111035890 Harga sampah organik: 20000 Harga sampah non organik: -	Muncul pemberitahuan bahwa harga sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
TC19	Create bank sampah dengan nama lokasi yang panjang sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Villageeeeeeeeeeee (e x260) Kontak Pengelola: 08111035890 Harga sampah organik: 20000 Harga sampah non organik: 30000	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid
TC20	create bank sampah dengan nama bank sampah yang panjang sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: a (x 256) Nama Pengelola: Jean Lokasi: Springvale Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: 15000	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid

TC21	Create bank sampah dengan nama yang sudah ada sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko Centre Lokasi: Qingce Villageee Kontak Pengelola: 081035890 Harga sampah organik: 12000 Harga sampah non organik: 13000	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid
TC22	Create bank sampah dengan nama bank sampah mengandung angka sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko24 Lokasi: Qingce Villageee Kontak Pengelola: 081035890 Harga sampah organik: 12000 Harga sampah non organik: 13001	Muncul Pemberitahuan nama Bank sampah hanya boleh berisi huruf dan minimal 3	Sesuai dengan ekspektasi	Valid
TC23	Create bank sampah dengan memasukkan harga negatif untuk harga sampah non-organik sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko24 Lokasi: Qingce Villageee Kontak Pengelola: 081035890 Harga sampah organik: 12000 Harga sampah non organik: -13002	Muncul pemberitahuan bahwa harga sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
TC24	Create bank sampah dengan memasukkan harga negatif untuk harga sampah organik sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko24 Lokasi: Qingce Villageee Kontak Pengelola: 081035890 Harga sampah organik: -12000 Harga sampah non organik: 13003	Muncul pemberitahuan bahwa harga sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
TC25	Create bank sampah dengan memasukkan harga negatif untuk harga sampah organik dan non-organik sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko24 Lokasi: Qingce Villageee Kontak Pengelola: 081035890 Harga sampah organik: -12000 Harga sampah non organik: -13004	Muncul pemberitahuan bahwa harga sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
TC26	Create bank sampah dengan data kontak lebih dari 15 digit angka sebagai penegiela	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liyue Ginko24 Lokasi: Qingce Villageee Kontak Pengelola: 081035890476759869486 Harga sampah organik: 12000 Harga sampah non organik: 13000	Muncul Pemberitahuan bahwa kontak harus diisi dengan angka minimal 7 dan maksimal 15	Sesuai dengan ekspektasi	Valid

TC27	Create bank sampah dengan data lokasi menggunakan angka sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: 4596464686 Kontak Pengelola: 080102030405 Harga sampah organik: 10000 Harga sampah non organik: 15000	Muncul Peringatan Bahwa Lokasi tidak boleh diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC28	Create bank sampah dengan data lokasi menggunakan angka sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liye Ginko24 Lokasi: 74574658764 Kontak Pengelola: 08103589047 Harga sampah organik: 12000 Harga sampah non organik: 13000	Muncul Peringatan Bahwa Lokasi tidak boleh diisi dengan angka	Sesuai dengan ekspektasi	Valid
TC29	Create bank sampah dengan data harga sampah organik lebih dari 255 karakter sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan nama pengelola 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: admin Password: admin  Nama bank sampah: Mondstadt Trash Bank Nama Pengelola: Jean Lokasi: Springval Kontak Pengelola: 080102030405 Harga sampah organik: 10000467653764986659676864369634674654766 6593687468563476587364583548357453583254 6853492534528345823585356354656532548345 8354653845238547523874582345823458234582345 832546235642348753284584875375287542853284 2546235642348753284584875375287542853284 58354845823565254818357254825487253481548 72538751487573451852732547235812458328751 873258715423578  Harga sampah non organik: 15000	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid
TC30	Create bank sampah dengan data harga sampah non-organik lebih dari 255 karakter sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Ketik 1 8. Masukkan nama bank sampah 9. Masukkan Lokasi 10. Masukkan kontak pengelola 11. Masukkan harga sampah organik 12. Masukkan harga sampah non-organik	Username: pengelola Password: pengelola  Nama bank sampah: Liye Ginko24 Lokasi: 74574658764 Kontak Pengelola: 08103589047 Harga sampah organik: 12000 Harga sampah non organik: 10000467653764986659676864369634674654766 6593687468563476587364583548357453583254 6853492534528345823585356354656532548345 8354653845238547523874582345823458234582345 832546235642348753284584875375287542853284 2546235642348753284584875375287542853284 58354845823565254818357254825487253481548 72538751487573451852732547235812458328751 873258715423578	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid
TC31	Create bank sampah sebagai pengelola yang belum memiliki bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 2 6. Ketik 1 7. Masukkan nama bank sampah 8. Masukkan Lokasi 9. Masukkan kontak pengelola 10. Masukkan harga sampah organik 11. Masukkan harga sampah non-organik 12. Enter	Username: LyoidHyper01 Password: iamSpy01  Nama bank sampah: Forgers Bank Lokasi: Kadipaten Brundhilde Kontak Pengelola: 0887987654 Harga sampah organik: 10000 Harga sampah non organik: 10000	Create bank sampah berhasil dan data tersimpan	Sesuai dengan ekspektasi	Valid

## Test Case Read

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
RD01	Read bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 1 11. Klik enter	1. Username: admin 2. Password: admin	Menampilkan data bank sampah	Sesuai dengan ekspektasi	Valid
RD02	Delete bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. Input nomor bank sampah 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan pesan data bank sampah berhasil dihapus	Sesuai dengan ekspektasi	Valid
RD03	Menghapus bank sampah yang tidak tersedia	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. Input nomor bank sampah 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan pesan data bank sampah tidak ada dalam pilihan	Sesuai dengan ekspektasi	Valid
RD04	Menghapus bank sampah dengan input symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. ketik @ 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan error message input hanya boleh berisi angka	Sesuai dengan ekspektasi	Valid
RD05	Menghapus bank sampah dengan input kosong	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. Klik enter	1. Username: admin 2. Password: admin	Menampilkan error message input tidak boleh kosong	Sesuai dengan ekspektasi	Valid
RD06	Menghapus bank sampah dengan input spasi	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. klik spasi 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan error message input tidak boleh kosong	Sesuai dengan ekspektasi	Valid
RD07	Menginput q ketika berada dihalaman delete bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 4 11. Klik enter 12. Ketik q 13. Klik enter	1. Username: admin 2. Password: admin	Kembali ke halaman pengelola	Sesuai dengan ekspektasi	Valid

## Test Case Edit

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
ED01	Edit bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter  12. Jalanakan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik q 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan data bank sampah, lalu memberikan input untuk mengedit bank sampah	Sesuai dengan ekspektasi	Valid
ED02	Menginput "q" ketika mengedit bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik q 13. Klik enter	1. Username: admin 2. Password: admin	Kembali ke halaman CRUD	Sesuai dengan ekspektasi	Valid
ED03	Menginput angka sesuai yang di tampilkan	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan output untuk mengubah data bank sampah	Sesuai dengan ekspektasi	Valid
ED04	Menginput dengan menggunakan huruf (kecuali q)	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik A 13. Klik enter	1. Username: admin 2. Password: admin	Memberikan alert message "Silahkan isi menggunakan angka"	Sesuai dengan ekspektasi	Valid
ED05	Mengubah nama bank sampah dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik @#\$@@ 15. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Nama bank sampah dan nama pengelola hanya boleh berisi huruf dan spasi! Minimal 3 huruf!"	Sesuai dengan ekspektasi	Valid
ED06	Mengubah lokasi bank sampah dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik *&\$@@ 19. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Lokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurung, setidaknya memiliki 3 huruf! Minimal 10 karakter!"	Sesuai dengan ekspektasi	Valid

ED07	Mengubah nomor telepon dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik #*\$@#\$!@\$ 21. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Kontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!"	Sesuai dengan ekspektasi	Valid
ED08	Mengubah harga dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik 0812975224 21. Klik enter 22. Ketik @\$#@# 23. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Harga sampah organik hanya boleh berisi angka, minimal 4 angka!"	Sesuai dengan ekspektasi	Valid
ED09	Mengubah nomor telepon dengan huruf	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik abcdefghij 21. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Kontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!"	Sesuai dengan ekspektasi	Valid
ED10	Mengubah harga dari angka ke huruf	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik 0812975224 21. Klik enter 22. Ketik abcdefg 23. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Harga sampah organik hanya boleh berisi angka, minimal 4 angka!"	Sesuai dengan ekspektasi	Valid
ED11	Update bank sampah dengan data kosong	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Klik enter 15. Klik enter 16. Klik enter 17. Klik enter 19. Klik enter	1. Username: admin 2. Password: admin	Data Tersimpan menampilkan message "Data bank sampah Anda berhasil diperbarui!"	Sesuai dengan ekspektasi	Valid

## Test Case Delete

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
ED01	Edit bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter	1. Username: admin 2. Password: admin	Menampilkan data bank sampah, lalu memberikan input untuk mengedit bank sampah	Sesuai dengan ekspektasi	Valid
ED02	Menginput "q" ketika mengedit bank sampah	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik q 13. Klik enter	1. Username: admin 2. Password: admin	Kembali ke halaman CRUD	Sesuai dengan ekspektasi	Valid
ED03	Menginput angka sesuai yang ditampilkan	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter	1. Username: admin 2. Password: admin	Menampilkan output untuk mengubah data bank sampah	Sesuai dengan ekspektasi	Valid
ED04	Menginput dengan menggunakan huruf (kecuali q)	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik A 13. Klik enter	1. Username: admin 2. Password: admin	Memberikan alert message "Silahkan isi menggunakan angka"	Sesuai dengan ekspektasi	Valid
ED05	Mengubah nama bank sampah dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik @#\$@ 15. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Nama bank sampah dan nama pengelola hanya boleh berisi huruf dan spasi! Minimal 3 huruf!"	Sesuai dengan ekspektasi	Valid

ED06	Mengubah lokasi bank sampah dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik * &@#@ 19. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Lokasi hanya boleh berisi huruf, spasi, angka, koma, spasi dan tanda kurang, setidaknya memiliki 3 huruf! Minimal 10 karakter!"	Sesuai dengan ekspektasi  Valid
ED07	Mengubah nomor telepon dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik #*\$@^\$!@#\$ 21. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Kontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!"	Sesuai dengan ekspektasi  Valid
ED08	Mengubah harga dengan symbol	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik 0812975224 21. Klik enter 22. Ketik \$@#@ 23. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Harga sampah organik hanya boleh berisi angka, minimal 4 angka!"	Sesuai dengan ekspektasi  Valid
ED09	Mengubah nomor telepon dengan huruf	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik abcdefghij 21. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Kontak hanya boleh berisi angka, minimal berisi 7 angka dan maksimal 15 angka!"	Sesuai dengan ekspektasi  Valid
ED10	Mengubah harga dari angka ke huruf	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Klik enter 6. Masukkan Password 7. Klik enter 8. Ketik 2 9. Klik enter 10. Ketik 3 11. Klik enter 12. Ketik 1 13. Klik enter 14. Ketik asolole 15. Klik enter 16. Ketik rehan 17. Klik enter 18. Ketik sukabirus 19. Klik enter 20. Ketik 0812975224 21. Klik enter 22. Ketik abcdefg 23. Klik enter	1. Username: admin 2. Password: admin	Data tidak tersimpan, memberikan alert message "Harga sampah organik hanya boleh berisi angka, minimal 4 angka!"	Sesuai dengan ekspektasi  Valid

## Test Case Search

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
SC01	Search berdasarkan nama bank sampah dengan data valid	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Nama Bank Sampah) 4. Masukkan nama bank sampah valid 5. Klik enter	Nama Bank Sampah: Sumera TrashBank	Menampilkan detail bank sampah dengan nama "Sumera TrashBank"	Sesuai dengan ekspektasi	Pass
SC02	Search berdasarkan nama bank sampah dengan data tidak valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Bank Sampah) 4. Masukkan nama bank sampah tidak valid 5. Klik enter	Nama Bank Sampah: Go Green	Sistem menampilkan pesan "Tidak ada hasil yang mendekati nama bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC03	Search berdasarkan nama pengelola bank sampah dengan data valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Masukkan nama pengelola valid 5. Klik enter	Nama Pengelola: Jakiii	Menampilkan detail bank sampah dengan nama pengelola "Jakiii"	Sesuai dengan ekspektasi	Pass
SC04	Search berdasarkan nama pengelola bank sampah dengan data tidak valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Masukkan nama pengelola tidak valid 5. Klik enter	Nama Pengelola: Luna	Sistem menampilkan pesan "Tidak ada hasil yang mendekati nama pengelola tersebut"	Sesuai dengan ekspektasi	Pass
SC05	Search berdasarkan lokasi dengan data valid	1. Jalankan program 2. Ketik 3 (Search bank Sampah) 3. Ketik 3 (Lokasi Bank Sampah) 4. Masukkan lokasi valid 5. Klik enter	Lokasi: Padang Pasir	Menampilkan detail bank sampah dengan lokasi "Padang Pasir"	Sesuai dengan ekspektasi	Pass
SC06	Search berdasarkan lokasi dengan data tidak valid	1. Jalankan program 2. Ketik 3 (Search bank Sampah) 3. Ketik 3 (Lokasi Bank Sampah) 4. Masukkan lokasi tidak valid 5. Klik enter	Lokasi: Sungai	Sistem menampilkan pesan "Tidak ada hasil yang mendekati lokasi bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC07	Search berdasarkan harga sampah organik dengan data valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Harga Sampah Organik) 4. Masukkan harga sampah organik valid 5. Klik enter	Harga Sampah Organik: 1000	Menampilkan detail bank sampah dengan nama pengelola "Jakiii"	Sesuai dengan ekspektasi	Pass

SC08	Search berdasarkan harga sampah organik dengan data tidak valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Harga Sampah Organik) 4. Masukkan harga sampah non organik tidak valid 5. Klik enter	Harga Sampah Organik: 6000	Sistem menampilkan pesan "Tidak ada hasil yang mendekati harga sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC09	Search berdasarkan harga sampah non organik dengan data valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 5 (Harga Sampah Non Organik) 4. Masukkan harga sampah organik valid 5. Klik enter	Harga Sampah Non Organik: 5000	Menampilkan detail bank sampah dengan nama pengelola "Jakiii"	Sesuai dengan ekspektasi	Pass
SC10	Search berdasarkan harga sampah non organik dengan data valid	1. Jalankan program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 5 (Harga Sampah Non Organik) 4. Masukkan harga sampah non organik tidak valid 5. Klik enter	Harga Sampah Non Organik: 4000	Sistem menampilkan pesan "Tidak ada hasil yang mendekati harga sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC11	Search berdasarkan nama bank sampah dengan kata kunci tidak lengkap	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Nama Bank Sampah) 4. Masukkan nama bank sampah dengan kata kunci tidak lengkap 5. Klik enter	Nama Bank Sampah: Nat	Menampilkan nama bank sampah yang mendekati dengan kata kunci	Sesuai dengan ekspektasi	Pass
SC12	Search berdasarkan nama pengelola dengan kata kunci tidak lengkap	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Masukkan nama pengelola dengan kata kunci tidak lengkap 5. Klik enter	Nama Pengelola: kii	Menampilkan nama pengelola yang mendekati dengan kata kunci	Sesuai dengan ekspektasi	Pass
SC13	Search berdasarkan lokasi dengan kata kunci tidak lengkap	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 3 (Lokasi Bank Sampah) 4. Masukkan lokasi dengan kata kunci tidak lengkap 5. Klik enter	Lokasi: dang	Menampilkan lokasi yang mendekati dengan kata kunci	Sesuai dengan ekspektasi	Pass
SC14	Search berdasarkan harga sampah organik dengan kata kunci tidak lengkap	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Harga Sampah Organik) 4. Masukkan harga sampah organik dengan kata kunci tidak lengkap 5. Klik enter	Harga Sampah Organik: 10	Menampilkan harga sampah yang relevan dengan kata kunci	Sesuai dengan ekspektasi	Pass
SC15	Search berdasarkan harga sampah non organik dengan kata kunci tidak lengkap	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 5 (Harga Sampah Non Organik) 4. Masukkan harga sampah non organik dengan kata kunci tidak lengkap 5. Klik enter	Harga Sampah Non Organik:	Menampilkan harga sampah yang relevan dengan kata kunci	Sesuai dengan ekspektasi	Pass
SC16	Search dengan data kosong pada bagian kategori	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Klik enter	Kategori:	Menampilkan peringatan "Silakan pilih kategori berdasarkan angka yang tersedia"	Sesuai dengan ekspektasi	Pass
SC17	Search berdasarkan nama bank sampah dengan data kosong	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Nama Bank Sampah) 4. Klik enter	Nama Bank Sampah:	Menampilkan peringatan "Input pencarian tidak boleh kosong!"	Sesuai dengan ekspektasi	Pass

SC18	Search berdasarkan nama pengelola dengan data kosong	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Klik enter	Nama Pengelola:	Menampilkan peringatan "Input pencarian tidak boleh kosong"	Sesuai dengan ekspektasi	Pass
SC19	Search berdasarkan lokasi dengan data kosong	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Lokasi Bank Sampah) 4. Klik enter	Lokasi:	Menampilkan peringatan "Input pencarian tidak boleh kosong"	Sesuai dengan ekspektasi	Pass
SC20	Search berdasarkan harga sampah organik dengan data kosong	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Harga Sampah Organik) 4. Klik enter	Harga Sampah Organik:	Menampilkan peringatan "Input pencarian tidak boleh kosong"	Sesuai dengan ekspektasi	Pass
SC21	Search berdasarkan harga sampah non organik dengan data kosong	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 5 (Harga Sampah Non Organik) 4. Klik enter	Harga Sampah Non Organik:	Menampilkan peringatan "Input pencarian tidak boleh kosong"	Sesuai dengan ekspektasi	Pass
SC22	Search dengan input simbol pada bagian kategori	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Masukkan simbol 4. Klik enter	Kategori: @!&\$	Menampilkan pesan "Silakan pilih kategori berdasarkan angka yang tersedia"	Sesuai dengan ekspektasi	Pass
SC23	Search berdasarkan nama bank sampah dengan input simbol	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Nama Bank Sampah) 4. Masukkan nama bank sampah dengan simbol 5. Klik enter	Nama Bank Sampah: @!&\$	Menampilkan pesan "Tidak ada hasil yang mendekati nama bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC23	Search berdasarkan nama pengelola dengan input simbol	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Masukkan nama pengelola dengan simbol 5. Klik enter	Nama Pengelola: @!&\$	Menampilkan pesan "Tidak ada hasil yang mendekati nama pengelola bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC24	Search berdasarkan lokasi dengan input simbol	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Lokasi Bank Sampah) 4. Masukkan lokasi dengan simbol 5. Klik enter	Lokasi: @!&\$	Menampilkan pesan "Tidak ada hasil yang mendekati lokasi bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC25	Search berdasarkan harga sampah organik dengan input simbol	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Harga Sampah Organik) 4. Masukkan harga sampah organik dengan simbol 5. Klik enter	Harga Sampah Organik: @!&\$	Menampilkan peringatan "Pencarian harga organik hanya boleh mengirim angka saja!"	Sesuai dengan ekspektasi	Pass

SC26	Search berdasarkan harga sampah non organik dengan input simbol	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Harga Sampah Non Organik) 4. Masukkan harga sampah non organik dengan simbol 5. Klik enter	Harga Sampah Non Organik: @!&\$	Menampilkan peringatan "Pencarian harga non organik hanya boleh mengirim angka saja!"	Sesuai dengan ekspektasi	Pass
SC27	Search dengan tipe data string pada bagian kategori	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Masukkan tipe data string 5. Klik enter	Kategori: Nama	Menampilkan peingatan "Silakan pilih kategori berdasarkan angka yang tersedia"	Sesuai dengan ekspektasi	Pass
SC28	Search berdasarkan nama bank sampah dengan angka	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 1 (Nama Bank Sampah) 4. Masukkan nama bank sampah dengan angka 5. Klik enter	Nama Bank Sampah: 834	Sistem menampilkan pesan "Tidak ada hasil yang mendekati nama bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC29	Search berdasarkan nama pengelola dengan angka	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 2 (Nama Pengelola Bank Sampah) 4. Masukkan nama pengelola dengan angka 5. Klik enter	Nama Pengelola: 834	Sistem menampilkan pesan "Tidak ada hasil yang mendekati nama pengelola bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC30	Search berdasarkan lokasi dengan angka	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 3 (Lokasi Bank Sampah) 4. Masukkan lokasi dengan angka 5. Klik enter	Lokasi: 834	Sistem menampilkan pesan "Tidak ada hasil yang mendekati lokasi bank sampah tersebut"	Sesuai dengan ekspektasi	Pass
SC31	Search berdasarkan harga sampah organik dengan tipe data string	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 4 (Harga Sampah Organik) 4. Masukkan harga sampah organik dengan tipe data string 5. Klik enter	Harga Sampah Organik: Sepuluh ribu	Menampilkan peringatan "Pencarian harga organik hanya boleh menginputkan angka saja"	Sesuai dengan ekspektasi	Pass
SC32	Search berdasarkan harga sampah non organik dengan tipe data	1. Jalankan Program 2. Ketik 3 (Search Bank Sampah) 3. Ketik 5 (Harga Sampah Non Organik) 4. Masukkan harga sampah non organik dengan tipe data string 5. Klik enter	Harga Sampah Non Organik: Lima ribu	Menampilkan peringatan "Pencarian harga non organik hanya boleh menginputkan angka saja"	Sesuai dengan ekspektasi	Pass

## Test Case Rating

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Pass/Fail
RT01	Memilih nomor bank sampah dengan data valid	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter	No Bank sampah: 1	Menampilkan perintah untuk memberikan rating "Silakan pilih nomor bank sampah yang ingin diberi rating"	Sesuai dengan ekspektasi	Valid
RT02	Memilih nomor bank sampah yang tidak terdaftar di data	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter	No Bank sampah: 4	Menampilkan perintah "Silakan pilih opsi yang tersedia" dan menampilkan kembali data sampah yang tersedia	Sesuai dengan ekspektasi	Valid
RT03	Memilih ID bank sampah dengan data kosong	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter	No Bank Sampah:	Menampilkan perintah "Silakan isi berdasarkan angka, hanya boleh berisi 1 angka" dan menampilkan kembali data sampah yang tersedia	Sesuai dengan ekspektasi	Valid
RT04	Memilih ID bank sampah dengan tipe data string	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan nilai 6. Klik enter	No Bank Sampah: Satu	Menampilkan perintah "Silakan isi berdasarkan angka, hanya boleh berisi 1 angka" dan menampilkan kembali data sampah yang tersedia	Sesuai dengan ekspektasi	Valid
RT05	Memilih ID bank sampah dengan simbol	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan nilai 6. Klik enter	No Bank Sampah: !5%	Menampilkan perintah "Silakan isi berdasarkan angka, hanya boleh berisi 1 angka" dan menampilkan kembali data sampah yang tersedia	Sesuai dengan ekspektasi	Valid
RT06	Memberikan rating untuk bank sampah dengan nilai valid	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan rating 6. Klik enter	No Bank Sampah: 1 Rating: 5	Rating berhasil disimpan dan menampilkan pesan "Terima kasih telah memberi penilaian untuk [nama Bank Sampah]"	Sesuai dengan ekspektasi	Valid
RT07	Memberikan rating untuk bank sampah dengan nilai tidak valid (di luar range)	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan rating 6. Klik enter	Rating: 9	Menampilkan perintah "Silakan rating bank sampah dari 1 sampai 5" dan menampilkan kembali data bank sampah	Sesuai dengan ekspektasi	Valid
RT08	Memberikan rating untuk bank sampah dengan data kosong	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan nilai 6. Klik enter	Rating:	Menampilkan perintah "Silakan rating bank sampah dari 1 sampai 5" dan menampilkan kembali data bank sampah	Sesuai dengan ekspektasi	Valid
RT09	Memberikan rating untuk bank sampah yang sudah dinilai sebelumnya	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter	Rating: 5	Data rating sebelumnya sudah tersimpan, sehingga tidak dapat memberikan rating kembali dan menampilkan pesan "Rating Anda untuk [nama bank sampah] adalah [rating]!"	Sesuai dengan ekspektasi	Valid
RT10	Memberikan rating untuk bank sampah dengan tipe data string	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan nilai 6. Klik enter	Rating: Dua	Menampilkan perintah "Silakan rating bank sampah dari 1 sampai 5" dan menampilkan kembali data bank sampah	Sesuai dengan ekspektasi	Valid
RT11	Memberikan rating untuk bank sampah dengan simbol	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Pilih bank sampah 4. Klik enter 5. Masukkan nilai 6. Klik enter	Rating: @!()	Menampilkan perintah "Silakan rating bank sampah dari 1 sampai 5" dan menampilkan kembali data bank sampah	Sesuai dengan ekspektasi	Valid
RT12	Membatalkan proses rating setelah memilih opsi rating bank sampah	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Ketik q (q untuk quit)	Pilihan: q	Kembali ke menu utama	Sesuai dengan ekspektasi	Valid
RT13	Membatalkan proses setelah memilih bank sampah yang akan diberikan rating	1. Jalankan Program 2. Ketik 4 (Rating Bank Sampah) 3. Ketik q (q untuk quit)	Pilihan: q	Kembali ke menu utama	Sesuai dengan ekspektasi	Valid

## Test Case EcosCalc

TestCase ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	PASS/FAIL
EC01	Melakukan Proses EcosCalc dengan data numerik valid sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik 2 11. Enter 12. Ketik 6 13. Enter	Username: admin Password: admin  Berat sampah Organik: 10 Berat sampah non-organik: 10	Proses EcosCalc berhasil dan menampilkan kisaran harga hasil perhitungan	Sesuai dengan ekspektasi	Valid
EC02	Melakukan Proses EcosCalc dengan data numerik valid sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik 2 11. Enter 12. Ketik 6 13. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: 10 Berat sampah non-organik: 10	Proses EcosCalc berhasil dan menampilkan kisaran harga hasil perhitungan	Sesuai dengan ekspektasi	Valid
EC03	Melakukan proses EcosCalc dengan data berat sampah organik negatif sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: admin Password: admin  Berat sampah Organik: -10 Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
EC04	Melakukan proses EcosCalc dengan data berat sampah non-organik negatif sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: 10 Berat sampah non-organik: -10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
EC05	Melakukan EcosCalc dengan menginputkan berat sampah organik dengan simbol sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: admin Password: admin  Berat sampah Organik: @\$%^#^# Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh diinput dengan simbol	Sesuai dengan ekspektasi	Valid
EC06	Melakukan EcosCalc dengan menginputkan berat sampah non-organik dengan simbol sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: 10 Berat sampah non-organik: @\$%^#^#	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh diinput dengan simbol	Sesuai dengan ekspektasi	Valid
EC07	Melakukan EcosCalc dengan Menginputkan berat kedua jenis sampah dengan simbol sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: admin Password: admin  Berat sampah Organik: @\$%^#^# Berat sampah non-organik: @@####	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh diinput dengan simbol	Sesuai dengan ekspektasi	Valid

EC08	Melakukan EcosCalc dengan menginputkan berat sampah organik lebih dari 255 karakter sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: 100004676537649866596768643696346746547666593 687468563476587364458354835745358325468534825 3452834582358353653465653254834583465384523 8547523874582352873474582358325462354625348 523465346532237645623456325462356425348753284 58487537528754285328458353458235652548183572 548254872534815487253875148757345185273254723 5812458328751873258715423578  Berat sampah non-organik: 10	Muncul Pemberitahuan bahwa EcosCalc hanya boleh 3 digit angka	Sesuai dengan ekspektasi	Valid
EC09	Melakukan EcosCalc dengan menginputkan berat sampah non-organik lebih dari 255 karakter sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: admin Password: admin  Berat sampah Organik: 10 Berat sampah non-organik: 100004676537649866596768643696346746547666593 687468563476587364458354835745358325468534825 3452834582358353653465653254834583465384523 8547523874582352873474582358325462354625348 523465346532237645623456325462356425348753284 58487537528754285328458353458235652548183572 548254872534815487253875148757345185273254723 5812458328751873258715423578	Muncul Pemberitahuan bahwa EcosCalc hanya boleh 3 digit angka	Sesuai dengan ekspektasi	Valid
EC10	Melakukan EcosCalc dengan berat sampah organik kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Kosongkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: - Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
EC11	Melakukan EcosCalc dengan berat sampah non-organik kosong sebagai admin	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Kosongkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: admin Password: admin  Berat sampah Organik: 10 Berat sampah non-organik: -	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
EC12	Melakukan EcosCalc dengan berat kedua jenis sampah kosong sebagai pengelola	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Kosongkan berat sampah organik 8. Kosongkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: pengelola Password: pengelola  Berat sampah Organik: - Berat sampah non-organik: -	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
EC13	Melakukan Proses EcosCalc dengan data numerik valid sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Mssukan berat sampah organik 8. Mssukan berat sampah non-organik 9. Enter 10. Ketik 2 11. Enter 12. Ketik 6 13. Enter	Username: user Password: user  Berat sampah Organik: 10 Berat sampah non-organik: 10	Proses EcosCalc berhasil dan menampilkan kisaran harga hasil perhitungan	Sesuai dengan ekspektasi	Valid
EC14	Melakukan proses EcosCalc dengan data berat sampah organik negatif sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Mssukan berat sampah organik 8. Mssukan berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: 10 Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
EC15	Melakukan proses EcosCalc dengan data berat sampah non-organik negatif sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Mssukan berat sampah organik 8. Mssukan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: -10 Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh negatif	Sesuai dengan ekspektasi	Valid
EC16	Melakukan proses EcosCalc dengan data berat sampah non-organik 0 (nol) sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Mssukan berat sampah organik 8. Mssukan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: 10 Berat sampah non-organik: 0	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh nol	Sesuai dengan ekspektasi	Valid

EC17	Melakukan proses EcosCalc dengan data berat sampah organik 0 (nol) sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: 0 Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh nol	Sesuai dengan ekspektasi	Valid
EC18	Melakukan proses EcosCalc dengan data berat sampah organik dan non-organik 0 (nol) sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: 0 Berat sampah non-organik: 0	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh nol	Sesuai dengan ekspektasi	Valid
EC19	Melakukan proses EcosCalc dengan data berat sampah organik kosong sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Kosongkan berat sampah organik 8. Masukkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: - Berat sampah non-organik: 10	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
EC20	Melakukan proses EcosCalc dengan data berat sampah non-organik kosong sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Masukkan berat sampah organik 8. Kosongkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: 10 Berat sampah non-organik: -	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid
EC21	Melakukan proses EcosCalc dengan data berat sampah organik dan non-organik kosong sebagai user	1. Jalankan program 2. Ketik 2 3. Masukkan Username 4. Masukkan Password 5. Ketik 1 6. Pilih Bank Sampah (Ketik 0) 7. Kosongkan berat sampah organik 8. Kosongkan Berat sampah non-organik 9. Enter 10. Ketik q 11. Enter	Username: user Password: user  Berat sampah Organik: - Berat sampah non-organik: -	Proses EcosCalc gagal dan menampilkan pemberitahuan berat sampah tidak boleh kosong	Sesuai dengan ekspektasi	Valid

## **Lampiran**

### **Lampiran 1 Wawancara dengan Stakeholder**

#### **Wawancara dengan User**

##### **1. Pertanyaan pertama**

Bagaimana pendapat Anda tentang sampah di sekitar?

User memberikan jawaban bahwa di sekitar UPI Cibiru terdapat banyak sampah sehingga tidak nyaman untuk di pandang. Kemudian user memberikan keluhannya mengenai kebingungannya untuk membuang sampah di mana dikarenakan kurangnya informasi mengenai lokasi disediakannya tempat pembuangan sampah di sekitar kampus UPI Cibiru

##### **2. Pertanyaan kedua**

Apa yang Anda inginkan agar masalah dapat terselesaikan ?

User memberikan pendapat seharusnya disediakan informasi mengenai lokasi tempat pembuangan sampah dalam jumlah besar. Sehingga memberikan kemudahan untuk membuang sampah dalam jumlah besar. Informasi tersebut memberikan manfaat kepada masyarakat lain yang memiliki permasalahan sama. Selain itu, dapat menarik minat masyarakat untuk membuang sampah pada tempat yang telah disediakan.

#### **Wawancara dengan Pengelola**

##### **1. Pertanyaan Pertama**

Fasilitas apa yang ada pada bank sampah yang anda kelola?

Di bank sampah ini terdapat fasilitas-fasilitas yang umumnya sudah ada pada bank sampah seperti penampungan, pemilahan sampah, penimbang dan lain-lain. Akan tetapi di sini terdapat fasilitas untuk penukaran sampah yang ditukarkan dengan nominal uang tertentu.

##### **2. Pernyataan Kedua**

Menurut anda, apa yang menyebabkan banyaknya sampah berserakan di lingkungan sekitar?

Menurut saya yang menyebabkan hal tersebut terjadi adalah kurangnya sosialisasi kepada masyarakat mengenai sampah, serta minimnya kepedulian terhadap kebersihan lingkungan pada masyarakat.

##### **3. Pertanyaan Ketiga**

Menurut anda bagaimana cara mengatasi hal tersebut?

Menurut saya, hal yang dapat dilakukan adalah memberikan sosialisasi kepada masyarakat, mulai dari pentingnya menjaga lingkungan sampai pada alur pembuangan sampah beserta lokasi-lokasi dari tempat penampungan yang ada.

#### Lampiran 2 Dokumentasi Wawancara



Gambar Lampiran 2 Dokumentasi Wawancara